Applied and Numerical Harmonic Analysis

David Joyner
Jon-Lark Kim

Selected Unsolved Problems in Coding Theory

Birkhäuser

Applied and Numerical Harmonic Analysis

David Joyner • Jon-Lark Kim

Selected Unsolved Problems in Coding Theory

Birkhäuser

David Joyner
Mathematics Department
US Naval Academy
Chauvenet Hall
Holloway Road 572C
Annapolis, Maryland 21402
USA
wdjoyner@gmail.com

Jon-Lark Kim
Department of Mathematics
University of Louisville
Natural Sciences Building 328
Louisville, KY 40292
USA
jl kim@louisville.edu

ISBN 978-0-8176-8255-2 e-ISBN 978-0-8176-8256-9
DOI 10.1007/978-0-8176-8256-9
Springer New York Dordrecht Heidelberg London

Library of Congress Control Number: 2011935547

Mathematics Subject Classification (2010): 94-02, 94B05, 94B25, 94B27, 11T71, 14G50, 05B05, 05B15

Printed on acid-free paper

www.birkhauser-science.com

Preface

This book is intended for research mathematicians interested in unsolved problems, and graduate students in mathematics or engineering who are interested in the mathematical side of the theory of error-correcting codes. It also may be of interest to coding-theorists who simply want to know how to use SAGE to do certain computations with error-correcting codes.

Strong undergraduates should find much in this book of some interest as well. In terms of classroom use, this text could serve as a basis for a special topics course in the theory of error-correcting codes. A good background in algebra, especially linear algebra, would be needed from the student. Some sections also require a strong background in algebraic geometry and number theory.

Coding theory is the branch of mathematics concerned with reliably transmitting data across noisy channels. In many cases, one can simply subdivide the data stream into blocks of a fixed *length* k and then encode each such block with some redundancy to a "codeword" of longer length n, which is then transmitted. With enough redundancy, the hope is that the receiver can recover the original k data bits. For example, in the late 1960s to early 1970s NASA's Mariner 9 took pictures[1] of Mars such as in Fig. 1. Black and white images such as in Fig. 1 were transmitted through space back to Earth using the so-called Reed–Muller code of length $n = 32$, with $k = 6$ data bits and $n - k = 26$ redundancy bits.

In spite of over 60 years of intensive work from the best minds in the world, there are many interesting mathematical questions which remain unsolved in the theory of error-correcting codes. The modest aim of this book will provide some "publicity" for some of those questions.

A chapter-by-chapter overview follows. We have tried to order the chapters by the rough level of mathematical sophistication required from the reader.

Chapter 1 contains a brief discussion of some basic terms and results on error-correcting codes. For example, the binary symmetric channel, entropy and uncer-

[1]This image of Mars' Olympus Mons was found on the NASA website http://marsprogram.jpl.nasa.gov/MPF/martianchronicle/martianchron2/index.html and is in the public domain. See also http://en.wikipedia.org/wiki/Mariner_9.

Fig. 1 Mars' Olympus Mons taken by Mariner 9

tainty, Shannon's theorem, the Hamming metric, the weight distribution (or spectrum) of a code, decoding basics, bounds on the parameters of a code (such as the Singleton bound, Manin's theorem, and the Gilbert–Varshamov asymptotic bound), and examples of important codes such as the Hamming codes and the quadratic residue codes. SAGE examples are scattered throughout to emphasize the computational aspect.

Chapter 2 is a very short chapter surveying certain aspects of the beautiful theory resulting in the intersection between self-dual codes, lattices, and invariant theory. This is a large field with several excellent books and survey articles already written. We introduce weight enumerator polynomials (and the MacWilliams identity), divisible codes and their classification, invariants associated to the different types of self-dual codes arising in this classification, and lattices arising from self-dual codes. The chapter ends with a discussion of the famous unsolved (at present) problem of the existence of a self-dual $[72, 36, 16]$ binary code. Again, some SAGE examples are given. A few proofs are sketched, but most results are stated with only references to original proofs.

Chapter 3 discusses some fascinating results in the intersection between coding theory, block designs, group theory, orthogonal arrays, Latin squares, and recreational mathematics. After introducing Hadamard matrices (and the Hadamard conjecture, with SAGE examples), one of the most remarkable results in all of coding theory is discussed, the Assmus–Mattson theorem. Roughly speaking, this theorem shows a relationship between certain codes and the construction of certain block designs. Connections with Latin squares and orthogonal arrays are given. The unexpected combinatorial structure "hidden" in certain "design-theoretic" codes is exemplified by the constructions in the section involving a Golay code and the "kitten" and "minimog" constructions. The last sections of the chapter discuss recreational aspects of the theory—strategies for winning a "mathematical blackjack" card-game and horsetrack-betting.

Chapter 4 explores an intriguing analogy between the Duursma zeta function (a recently introduced "invariant" object associated to a linear code) and the zeta function attached to an algebraic curve over a finite field. Much remains unknown (at this time) regarding the Duursma zeta function, but this chapter surveys its known properties (mostly with proofs). Several SAGE examples are given; in fact, SAGE is the only mathematics software package (at this time) with commands to compute Duursma zeta functions.

Chapter 5 discusses two very hard and unsolved problems. The first is a nontrivial estimate for the number of solutions (mod p) to a polynomial equation $y^2 = f(x)$, where $f(x)$ is a polynomial whose degree is "small" compared to the prime p. (When p is small compared to the degree of f, then Weil's estimate gives a good estimate of the number of solutions.) The second unsolved problem is the best asymptotic bounds for a binary linear code. The surprise is that these two seemingly unrelated problems are in fact, rather closely related. Aspects of this relationship, with some proofs and SAGE examples, are discussed in detail.

Finally, Chap. 6 discusses some aspects of algebraic-geometric codes (or AG codes, for short). These are codes arising generally from algebraic varieties over finite fields, though the focus here is on modular curves. This is a relatively technical chapter, requiring some familiarity of number theory, algebraic geometry, and modular forms and also of representation theory of finite groups. Fitting with the general theme of this book, this chapter mostly illustrates how little we know about the algebraic structure of AG codes arising from modular curves. As with many other areas of mathematics, it seems that the more one knows, the more one discovers how little is really known.

Acknowledgements DJ: I thank John Benedetto for the suggestion to write this book and all his encouragement over the years.

JLK: I thank Professor Emeritus Vera Pless of University of Illinois at Chicago for teaching me the insight of coding theory. I also thank my coauthor David Joyner for his encouragement.

For Chap. 3, we thank Alex Ryba and Andy Buchanan for helpful comments, and Ann Casey, who coauthored (with DJ) a much earlier and shorter version.

For Chap. 4, we are grateful to Thann Ward for the reference to [Sl], Koji Chinen for many interesting emails about his work, and to Cary Huffman and Iwan Duursma for very interesting conversations on this topic.

For Chap. 5, we thank Prof. Amin Shokrollahi of the Ecole Polytechnique Fédérale de Lausanne for helpful advice and Prof. Felipe Voloch of the University of Texas for allowing his construction to be included above. Parts of this (such as Proposition 156) can be found in the honors thesis [C] of DJ's former student Greg Coy, who was a pleasure to work with.

Part of Chap. 6 was written with Salahoddin Shokranian of the Universidade de Brasília (and Amin Shokrollahi's uncle!). Other parts were adapted from a paper written with Amy Ksir (of the US Naval Academy). We also thank D. Prasad and R. Guralnick for enlightening correspondence and in particular for the references [KP] and [Ja1].

Contents

Chapter 1
Background on Information Theory and Coding Theory

This chapter summarizing background information assumes that the reader has some familiarity with linear algebra and basic probability. The basic model of information theory and error-correcting block codes is introduced. The basic example of the Hamming $[7, 4, 3]$ code is presented in detail.

What is ironic is that even in basic background issues, coding theory has interesting open questions. For example, for a given length and dimension, which code is the best 2-error-correcting code? Another example: see Manin's theorem 19 and the closely related Conjecture 22 below.

1.1 Binary Symmetric Channel

Consider a source sending messages through a noisy channel. For example, consider a CD player reading from a scratched music CD, or a wireless cell phone capturing a weak signal from a relay tower which is too far away. These situations give rise to problems in communication which must be solved if one is to transmit information reliably. Figure 1.1 gives a diagram of the basic idea.

For simplicity, assume that the message being sent is a sequence of 0s and 1s. Assume that, due to noise, the probability that a 0 is (correctly) received is $1 - p$ and the probability that a 1 is (incorrectly) received is p. Assume also that the noise of the channel is not dependent on the symbol sent: the probability that a 1 is (correctly) received is $1 - p$, and the probability that a 0 is (incorrectly) received is p. This channel is called the *binary symmetric channel*.

The "butterfly" diagram in Fig. 1.2 summarizes this.

We also assume that the probability of an error in transmitting a bit does not depend on any previous transmission. This is called the *memoryless* binary symmetric channel.

D. Joyner, J.-L. Kim, *Selected Unsolved Problems in Coding Theory*,
Applied and Numerical Harmonic Analysis,
DOI 10.1007/978-0-8176-8256-9_1,

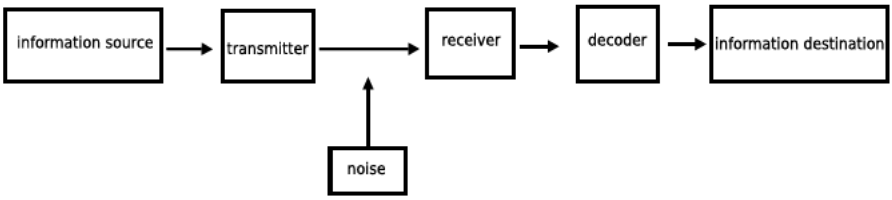


Fig. 1.1 The basic communication model

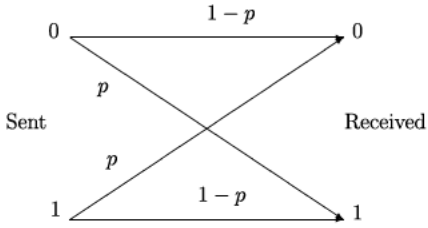


Fig. 1.2 The binary symmetric channel

1.1.1 Uncertainty

We want to formalize the notion of uncertainty. Consider two experiments. In the first, you flip a fair coin. In the second, you roll a fair dice. It is reasonable to say that the outcome of the second experiment is more uncertain than the outcome of the first, simply because there are more possibilities to choose from.

Suppose that a random variable X takes on only the distinct values $x_1, \ldots, x_n$ with nonzero probabilities $p_1, \ldots, p_n$, resp., where $p_1 + \cdots + p_n = 1$, and each $p_i \in [0, 1]$. How would you measure quantitatively the "randomness" or "uncertainty" of X? Claude Shannon, motivated by ideas of Norbert Wiener [CSi], introduced the following definition.

Definition 1 The *uncertainty* or (base 2) *entropy* of the above random variable X is defined to be

$$H(X) = H(p_1, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log_2 p_i,$$

where $\log_2$ is the logarithm to the base 2. The *base q entropy* is the same quantity, but the logarithm is taken with base q:

$$H_q(X) = H_q(p_1, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log_q p_i.$$

The entropy is always nonnegative and is only 0 when one of the p_i's is equal to 1. Its maximum value, $\log_2 n$, is only attained when all the p_i's are equal to $1/n$. Figure 1.3 has an example of this function with base 2 and $n = 2$.

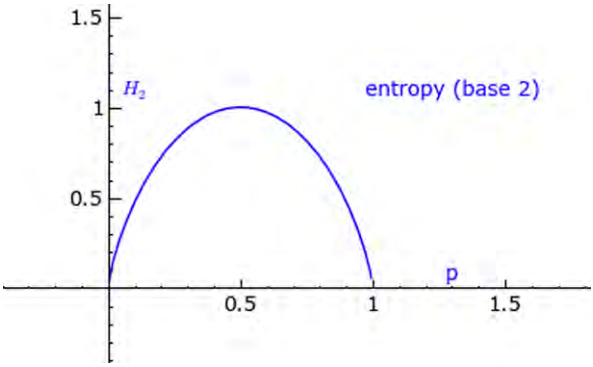


Fig. 1.3 The base 2 entropy function

Example 2 Suppose that X is the signal received by a (noisy) binary symmetric channel. Then

$$H(X) = H(p, 1-p) = -p\log_2 p - (1-p)\log_2(1-p).$$

The maximum uncertainty is when $p = 1/2$ (e.g., tossing a fair coin), in which case $H(X) = 1$. If $p = 0.99$, then $H(X) = 0.080793\ldots$.

It is intuitively obvious that when the channel creates a lot of errors, then there is a limitation to the information which can be sent. The next definition makes this idea more precise.

Definition 3 The *capacity* (or Shannon capacity) of the channel is $\text{cap}(X) = \text{cap}(p) = 1 - H(X)$.

In the case of the binary symmetric channel, the capacity is $\text{cap}(X) = 1 + p\log_2 p + (1-p)\log_2(1-p)$. The minimum capacity is when $p = 1/2$.

To justify the formula which defines the capacity, we need Shannon's fundamental theorem of information theory (also called the noisy channel theorem, see Sect. 1.1.2 below). However, recent research shows that one can explicitly construct codes which nearly achieve the Shannon capacity ([BGT, MN, VVS]). In fact, some have said that this construction marks "the death of coding theory" (for example, D. Forney's 1995 Shannon Lecture; see [Af] for more history). However, it appears that research in the topic has not abated, and if the conjectures in this book are any indication, then there are still plenty of interesting open questions left in coding theory!

1.1.2 Shannon's Theorem

The following theorem of Shannon is fundamental to the theory of error-correcting codes.

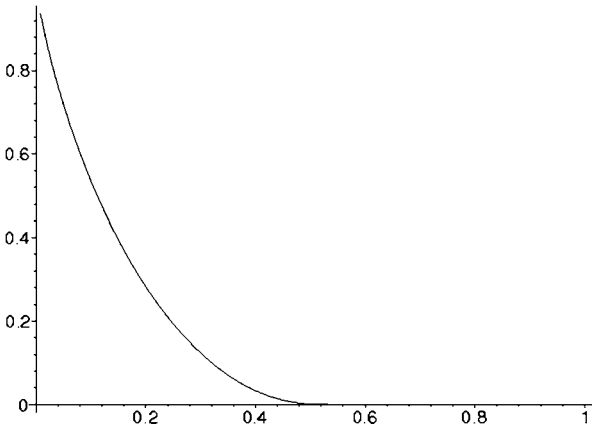


Fig. 1.4 The capacity function for binary codes

Theorem 4 (Fundamental theorem of information theory) *Consider a binary symmetric channel with* $p < 1/2$. *Let* $\epsilon > 0$ *and* $\delta > 0$ *be given. For all sufficiently large* n, *there is a code* $C \subset GF(2)^n$ *with information rate* R *satisfying* $\text{cap}(p) - \epsilon < R < \text{cap}(p)$, *such that the nearest neighbor algorithm decoding has average probability of incorrect decoding less than* δ.

Shannon's theorem guarantees us that "good" (in the sense that they are close to being best possible) codes exist. They may not be linear and even if they are, the theorem does not suggest that they are practical (i.e., "fast" encoding and decoding algorithms exist). The proof of Shannon's theorem is not easy and goes beyond the scope of this book. For a proof and further discussion, see Ash [Ash], Sect. 3.5 or van Lint [vL1], Chap. 2.

Figure 1.4 graphs the capacity for a binary code.

1.2 A Simple Example

Historically, the first error-correcting code to arise was the so-called binary Hamming $[7, 4, 3]$ code, discovered by Richard Hamming[1] in the late 1940s. Hamming, who worked for many years as a mathematician for a telephone company, thought that computers should be able to correct bit errors. He was right and discovered an infinite family of 1-error correcting codes, now called "Hamming codes." By way of

[1]From the publication point of view, Hamming published only binary $[7, 4, 3]$ code, and Golay published the other binary and nonbinary Hamming codes. However it has been shown that Hamming knew all the binary codes prior to Shannon's publication and had circulated them in an interdepartmental memorandum several months prior to the submission date of Golay's one-page paper [Tho].

introducing ideas such as block length, redundancy, and error-correction, we shall focus briefly only on one of them and define the more general codes later.

Let $\mathbb{F} = GF(2)$, $k = 4$, $n = 7$, and let C be the set of all vectors in the third column below (for simplicity, we write 0000000 instead of $(0, 0, 0, 0, 0, 0, 0)$, for example).[2]

Decimal	Binary "data"	Codewords
0	0000	0000000
1	0001	0001110
2	0010	0010101
3	0011	0011011
4	0100	0100011
5	0101	0101101
6	0110	0110110
7	0111	0111000
8	1000	1000111
9	1001	1001001
10	1010	1010010
11	1011	1011100
12	1100	1100100
13	1101	1101010
14	1110	1110001
15	1111	1111111

This is a linear code of length 7, dimension 4, and minimum distance 3 and is called the *Hamming* $[7, 4, 3]$*-code*. In fact, there is an "encoding" map from $\mathbb{F}^4$ to C given by $\phi(x) = y$, where

$$y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \pmod 2 = \phi(x),$$

[2]Here "*GF*" stands for Galois field, named after the French mathematician Evariste Galois who died after a duel at the age of 20. See http://en.wikipedia.org/wiki/Evariste_Galois for more details on his life's story.

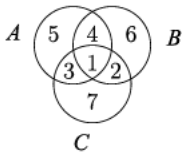


Fig. 1.5 Venn diagram for decoding a Hamming [7, 4, 3] code

and ϕ may be identified with the above 7×4 matrix. A basis for C is given by the vectors

$$\phi(e_1) = (1,0,0,0,1,1,1), \qquad \phi(e_2) = (0,1,0,0,0,1,1),$$
$$\phi(e_3) = (0,0,1,0,1,0,1), \qquad \phi(e_4) = (0,0,0,1,1,1,0),$$

where e_1, e_2, e_3, e_4 are the basis (column) vectors of $GF(2)^4$. Therefore, the rows of the transpose matrix of ϕ,

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix},$$

form a basis for C, i.e., G is a generator matrix. Another way to define C is to say that a vector $v \in GF(2)^7$ belongs to C if and only if it satisfies the "parity check conditions" $Hv = 0$, where

$$H = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Next, we present a simple algorithm showing how the three bits of redundancy can be used to decode a received codeword in which the noise of the communication channel has introduced an error in one of the 7 bits of the codeword which was transmitted.

An Algorithm for Decoding the Hamming [7, 4, 3] Code Denote the received word by $w = (w_1, w_2, w_3, w_4, w_5, w_6, w_7)$.

1. Put w_i in region i of the Venn diagram in Fig. 1.5, $i = 1, 2, \ldots, 7$.
2. Do parity checks[3] on each of the circles A, B, and C.

[3] In other words, add the entries placed in each circle mod 2. If this sum is $\equiv 1 \pmod 2$, then we say that the circle *fails* the parity check; otherwise, it *passes*. See Example 5.

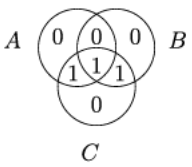


Fig. 1.6 Venn diagram for decoding a Hamming [7, 4, 3] code with error positions filled in

Parity failure region(s)	Error position
none	none
A, B, and C	1
B and C	2
A and C	3
A and B	4
A	5
B	6
C	7

These rules cleverly reformulate the algebraic "parity check conditions" $Hv = 0$ as a more visually appealing Venn diagram.

Example 5 Consider the binary Hamming code C above. Suppose that $v = (1, 1, 1, 0, 0, 0, 0)$ was received from a transmission along a noisy channel. What was the codeword (most likely) sent?

The corresponding Venn diagram is given in Fig. 1.6.

If we do parity checks on each of the circles A, B, and C, we see that only C fails. The table tells us that an error occurred in the 7th bit; therefore, $v = (1, 1, 1, 0, 0, 0, 0)$ decodes to $c = (1, 1, 1, 0, 0, 0, 1)$.

This above example raises a number of questions.

- Are there other codes of length $n = 7$ and dimension $k = 4$ which can correct more errors? In other words, is C the "best" we can do for that n and k? (No and Yes.)
- Are there other decoding algorithms? (Yes, in case you do not like Venn diagrams!)
- Does this example generalize? (Yes, in several ways.)

1.3 Basic Definitions

Enough examples; now for the definition. Let $\mathbb{F} = GF(q)$ be a finite field.[4]

[4]The appendix Sect. 7.2 gives further details on finite fields.

Definition 6 A subset C of $V = \mathbb{F}^n$ is called a *code of length* n. If C has M elements, then it is often referred to as an (n, M)*-code*. A subspace of V is called a *linear code of length* n. If $\mathbb{F} = GF(2)$, then C is called a *binary* code. If $\mathbb{F} = GF(3)$, then C is called a *ternary* code. In general, when $\mathbb{F} = GF(q)$, then C is called a q*-ary* code. The elements of a code C are called *codewords*. For each vector $v \in V$, let

$$\operatorname{supp}(v) = \{i \mid v_i \neq 0\}$$

denote the *support* of the vector.

The *information rate* of C is

$$R = \frac{\log_q |C|}{n},$$

where $|C|$ denotes the number of elements of C.

Not only do we consider $V = \mathbb{F}^n$ as a vector space with a fixed basis but, in fact, as a vector space with a fixed basis and a fixed inner product. Unless stated otherwise, we give V the Euclidean inner product.

If $\cdot$ denotes the usual (Euclidean) inner product,

$$v \cdot w = v_1 w_1 + \cdots + v_n w_n,$$

where $v = (v_1, \ldots, v_n) \in V$ and $w = (w_1, \ldots, w_n) \in V$, then we define the *dual code* $C^\perp$ by

$$C^\perp = \{v \in V \mid v \cdot c = 0 \ \forall c \in C\}.$$

We say C is *self-dual* if $C = C^\perp$.

If q is a square, say $q = p^2$, then there is a conjugation on $\mathbb{F} = GF(q)$ (i.e., a field automorphism $\mathbb{F} \to \mathbb{F}$ of order 2 which fixes the subfield $GF(p)$), which we denote by $x \longmapsto \overline{x}$ for $x \in \mathbb{F}$.

If the order of $\mathbb{F}$ is a square and if $\langle v, w \rangle$ denotes a Hermitian inner product,

$$\langle v, w \rangle = \sum_{i=1}^{n} v_i \overline{w_i},$$

then the *Hermitian dual code* of C is

$$C^\perp = \{y \in V^n \mid \langle v, c \rangle = 0 \text{ for all } c \in C\}.$$

We say that C is *Hermitian self dual* if C is equal to its Hermitian dual code $C^\perp$.

The *conjugate code* of a code C over $GF(p^2)$ is the code of conjugates: $\overline{C} = \{\overline{c} \mid c \in C\}$. Note that the Hermitian dual code is the conjugate code of the Euclidean dual code.

1.3.1 The Hamming Metric

We have seen so far some simple examples of codes. What is needed is some notion of how to compare codewords. Geometrically, two codewords are "far" from each other if there are "a lot" of coordinates where they differ. This notion is made more precise in the following definition.

Definition 7 If $v = (v_1, v_2, \ldots, v_n)$, $w = (w_1, w_2, \ldots, w_n)$ are vectors in $V = \mathbb{F}^n$, then we define

$$d(v, w) = \left|\left\{i \mid 1 \le i \le n,\ v_i \ne w_i\right\}\right|$$

to be the *Hamming distance* between v and w. The function $d : V \times V \to \mathbb{N}$ is called the *Hamming metric*. If $v \in V$ and $S \subset V$, then we define the *distance from v to S* by

$$d(v, S) = \min_{s \in S} d(v, s).$$

The *weight* of a vector (in the Hamming metric) is $d(v, 0)$. The *weight distribution vector* or *spectrum* of a code $C \subset \mathbb{F}^n$ is the vector

$$A(C) = \operatorname{spec}(C) = [A_0, A_1, \ldots, A_n]$$

where $A_i = A_i(C)$ denotes the number of codewords in C of weight i for $0 \le i \le n$. Note that for a linear code C, $A_0(C) = 1$, since any vector space contains the zero vector.

Note that

$$d(v, w) = \left|\left\{i \mid 1 \le i \le n,\ v_i - w_i \ne 0\right\}\right| = d(v - w, 0) \tag{1.3.1}$$

for any vectors $v, w \in \mathbb{F}^n$ (or, more generally, any vectors in a linear code). Using this, it is easy to show that d satisfies the properties of a metric:

- $d(v, w) \ge 0$ for all $v, w \in \mathbb{F}^n$ and $d(v, w) = 0$ if and only if $v = w$.
- $d(v, w) = d(w, v)$ for all $v, w \in \mathbb{F}^n$.
- $d(u, w) \le d(u, v) + d(v, w)$ for all $u, v, w \in \mathbb{F}^n$.

For $v \in \mathbb{F}^n$, let

$$B\left(v, r, \mathbb{F}^n\right) = \left\{w \in \mathbb{F}^n \mid d(v, w) \le r\right\}.$$

This is called the *ball of radius r about v*. Since $\mathbb{F}^n$ is finite, this ball has only a finite number of elements. It is not hard to count them using a little bit of basic combinatorics. Since this count shall be needed later, we record it in the following result.

Lemma 8 *If $0 \le r \le n$ and $q = |\mathbb{F}|$, then*

$$\left|B\left(v, r, \mathbb{F}^n\right)\right| = \sum_{i=0}^{r} \binom{n}{i} (q-1)^i.$$

Proof Let

$$B_i\big(v, r, \mathbb{F}^n\big) = \big\{w \in \mathbb{F}^n \mid d(v, w) = i\big\}.$$

This is called the *shell of radius i about v*. It consists of all vectors with exactly i coordinates different from v. There are $\binom{n}{i}$ ways to choose i out of n coordinates. There are $(q-1)^i$ ways to choose these i coordinates to be different from those in v. Thus,

$$\big|B\big(v, r, \mathbb{F}^n\big)\big| = \sum_{i=0}^{r} \big|B_i\big(v, r, \mathbb{F}^n\big)\big| = \sum_{i=0}^{r} \binom{n}{i} (q-1)^i.$$

□

1.4 Linear Block Codes

A linear code of length n and dimension k (as a vector space over $\mathbb{F}$) is called an $[n, k]$-*code*. In abstract terms, an $[n, k]$-code is given by a short exact sequence[5]

$$0 \to \mathbb{F}^k \xrightarrow{G} \mathbb{F}^n \xrightarrow{H} \mathbb{F}^{n-k} \to 0. \tag{1.4.1}$$

We usually identify C with the image of G and regard C as a subspace of $\mathbb{F}^n$. The function

$$E : \mathbb{F}^k \to C,$$
$$m \longmapsto mG$$

(regarding m as a row vector) is called the *encoder* (or *encoding matrix*, which is the transpose of G). Since the sequence (1.4.1) is exact, a vector $v \in \mathbb{F}^n$ is a codeword if and only if $H(v) = 0$. If $\mathbb{F}^n$ is given the usual standard vector space basis, then the matrix of G is a *generator matrix* of C, and the matrix of H is a *check matrix* of C. In other words,

$$\begin{aligned} C &= \big\{c \mid c = mG, \text{ some } m \in \mathbb{F}^k\big\} \\ &= \big\{c \in \mathbb{F}^n \mid H \cdot {}^t c = 0\big\}, \end{aligned}$$

where ${}^t c$ is a column vector. When G has the block matrix form

$$G = (I_k \mid A),$$

where I_k denotes the $k \times k$ identity matrix, and A is some $k \times (n-k)$ matrix, then we say that G is in *standard form*. By abuse of terminology, if this is the case, then we say that C is in *standard form*.

[5] "Short exact" means (a) the arrow G is injective, i.e., G is a full-rank $k \times n$ matrix, (b) the arrow H is surjective, and (c) $\text{image}(G) = \text{kernel}(H)$.

The matrix G has rank k, so the row-reduced echelon form of G, call it G', has no rows equal to the zero vector. In fact, the standard basis vectors $e_1, \ldots, e_k$ of the column space $\mathbb{F}^k$ occur amongst k columns of those of G'. The corresponding coordinates of C are called the *information coordinates* (or information bits if C is binary) of C.

Remark 1 Aside: "Generically" a square matrix with real entries is invertible. In the case of finite fields, this is not the case. For example, the probability that a "large random" $k \times k$ matrix with entries in $GF(q)$ is invertible is about

$$\lim_{k\to\infty} \frac{(q^k-1)(q^k-q)\ldots(q^k-q^{k-1})}{q^{k^2}} = \prod_{i=1}^{\infty}(1-q^{-i}).$$

If $q = 2$, then this is about $0.288\ldots$; for $q = 3$, this is about 0.56; for $q = 4$, this is about 0.688; and for $q = 5$, this is about 0.76.

For more interesting facts like these, see Lecture 7 in A. Barg's EENEE 739C course (online [Ba]).

Note that the action of the symmetric group of degree n, S_n, on $\{1, 2, \ldots, n\}$ induces an action on $\mathbb{F}^n$ via its action the coordinates of each element of $\mathbb{F}^n$.

Example 9 If $\mathbb{F} = GF(11)$ and $V = \mathbb{F}^{10}$, then

$$C = \left\{(x_1, x_2, \ldots, x_{10}) | x_i \in \mathbb{F}, x_1 + 2x_2 + 3x_3 + \cdots + 9x_9 + 10x_{10} \equiv 0 \pmod{11}\right\}$$

is called the *ISBN code*. This is an 11-ary linear code of length 10. This is the same code used in book numbering except that the number 10 is denoted by X on the inside cover of a book.

For example, $(1, 0, 0, 0, 0, 0, 0, 0, 0, 1)$ and $(1, 1, 1, 1, 1, 1, 1, 1, 1, 1)$ are codewords. Their Hamming distance is 8.

Example 10 The US Post Office puts a bar code on each letter to help with its delivery. What are these funny symbols? Translated into digits, they are given in the table in Fig. 1.7.

Each "word" in the postal bar-code has 12 digits, each digit being represented by short bars (we regard as a 0) and longer bars (which are regarded as a 1), as above. The 12 digits are interpreted as follows: The first 5 digits are your zip code, the next 4 digits are the extended zip code, the next 2 digits are the delivery point digits, and the last digit is a check digit (all the digits must add to 0 mod 10).

For example, suppose that you had 5 bars you could not read, ?????, followed by

In other words, after translating the bars into digits, you found that the postal code on an envelope was

?62693155913,

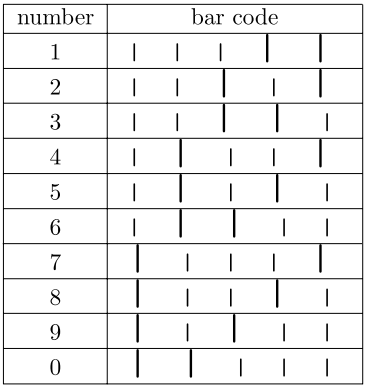


Fig. 1.7 US Post Office bar codes

where ? indicates a digit which was smudged so you could not read it. Since the sum must be $\equiv 0 \pmod{10}$, we must have $? = 0$.

1.4.1 Decoding Basics

If you have sent off a codeword $c \in C$ and the receiver received a vector $v \in \mathbb{F}^n$ which has some errors, then it is clear that the receiver wants to somehow recover c from v, if possible. This process is called (*error-correction*) *decoding*.

One approach is simply to brute force search for the codeword closest (in the sense of the Hamming distance) to the received vector. In this case, to decode the received vector v, all the receiver has to do is search C (which is a finite set) and find a codeword c' which is as close as possible to v (if there are several, then pick one at random). In most realistic situations, it is likely (in a sense that can be made precise) that $c' = c$. This strategy is called the *nearest neighbor algorithm*. Here is an algorithm implementing this strategy:

1. Input: A received vector $v \in \mathbb{F}^n$.
   Output: A codeword $c \in C$ closest to v.
2. Enumerate the elements of the ball $B_e(v)$ about the received word. Set $c =$ "fail".
3. For each $w \in B_e(v)$, check if $w \in C$. If so, put $c = w$ and break to the next step; otherwise, discard w and move to the next element.
4. Return c.

Note that "fail" is not returned unless $e > [\frac{d-1}{2}]$, where d denotes the minimum distance of C, and $[x]$ denotes the integer part of a real number $x > 0$.

The above algorithm has worst-case complexity exponential in n.

Definition 11 We say that a linear C is *e-error correcting* if $|B_e(w) \cap C| \leq 1$ for any $w \in \mathbb{F}^n$.

Here is another way to think of this definition. Assume that $C \subset \mathbb{F}^n$ is a linear code having minimum distance $d \geq 3$ and that $e \geq 1$ satisfies the property that for

each $w \in \mathbb{F}^n$ whose distance to C is less than or equal to e, there is a *unique* $c = c(w) \in C$ realizing this minimum: $d(w, c) = d(w, C) \leq e$. Therefore, C is e-error correcting if the following property holds: given any $w \in \mathbb{F}^n$ whose distance to C is $\leq e$, there is a unique $c' \in C$ which satisfies $d(w, c') \leq e$, and this c' is the codeword $c(w)$ described above. In other words, if a codeword at most distance e from w exists, then it is unique.

Suppose that C is e-error correcting. If you have sent off a codeword $c \in C$ and *know* that the receiver received a vector $v \in \mathbb{F}^n$ which has e errors or less (i.e., $d(c, v) \leq e$), then the codeword closest to v in the sense of the Hamming distance is c.

Example 12 Consider the binary code C with generator matrix

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}.$$

Suppose that $v = (1, 1, 1, 0, 0, 0, 0)$ was received from a transmission along a noisy channel. What was the codeword (most likely) sent?

Using SAGE, this can be done easily.

SAGE

```
sage: MS = MatrixSpace(GF(2),4,7)
sage: G = MS([[1,0,0,0,1,1,1],[0,1,0,0,0,1,1],\
 [0,0,1,0,1,0,1],[0,0,0,1,1,1,0]])
sage: C = LinearCode(G)
sage: V = VectorSpace(GF(2),7) # or V = GF(2)^7
sage: v = V([1,1,1,0,0,0,0])
sage: C.decode(v)
(1, 1, 1, 0, 0, 0, 1)
```

Therefore, $v = (1, 1, 1, 0, 0, 0, 0)$ decodes to $c = (1, 1, 1, 0, 0, 0, 1)$.

Definition 13 Let $\mathbb{F}$ be a finite field. A (possibly nonlinear) code $C \subset \mathbb{F}^n$ is called an (n, M, d)-*code* if it has length n, cardinality $|C| = M$, and minimum distance d. A linear code $C \subset \mathbb{F}^n$ is called an $[n, k, d]$-*code* if it has length n, dimension k, and minimum distance d.

For example, the ISBN code is a $[10, 9, 2]$-code over the field $GF(11)$.

It is easy to show that if d denotes the minimum distance of C, then the largest e for which G is e-error correcting is the integer part of $\frac{d-1}{2}$. Suppose that you have an $[n, k, d]$-code C and create a ball of radius $[\frac{d-1}{2}]$ about each codeword $c \in C$. Each vector $v \in V = GF(q)^n$ is either contained in exactly one of these balls or in none of them (in other words, the balls are disjoint; think of the gaps between a stack of oranges you might see in a grocery store).

1.4.2 Hamming Codes over $GF(q)$

The construction of the $[7, 4, 3]$-code mentioned earlier in the preface can be generalized considerably.

Let q be a prime power, $\mathbb{F} = GF(q)$, and let $\mathbb{F}^r - \{0\}$ denote the set of all nonzero r-tuples, i.e., the nonzero vectors in the r-dimensional vector space $\mathbb{F}^r$ over $\mathbb{F}$. Pick a bijective set-theoretic map, call it ψ, sending each vector in $\mathbb{F}^r$ to an r-tuple of integers in $Z_q = \{0, 1, \ldots, q-1\}$. We only require that ψ send the zero vector to the zero r-tuple. If $z \in \mathbb{F}^r - \{0\}$ and $\psi(z) = (a_1, \ldots, a_r) \in Z_q^r - \{0\}$, $0 \le a_i \le q-1$, then define

$$x(z) = a_1 + a_2 q + \cdots + a_r q^{r-1}.$$

This is a map $x : \mathbb{F}^r - \{0\} \to \{0, 1, \ldots, q^r - 1\}$. Let $z, z' \in \mathbb{F}^r - \{0\}$ be two such vectors. Define $z <_x z'$ if $x(z) < x(z')$. For each vector $z \in \mathbb{F}^r - \{0\}$, divide z by its first nonzero entry. Let S be the set of these "scaled" vectors. There are $n = (q^r - 1)/(q-1)$ elements in S. Write the elements of the set S in increasing order, using the ordering $<_x$ above,

$$S = \{s_1, \ldots, s_n\}. \tag{1.4.2}$$

Let H be the $r \times n$ matrix whose ith column is the ith vector in S (written as a column vector).

Example 14 For example, if $p = 3$ and $r = 2$, then

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 2 & 1 \end{pmatrix}$$

is a parity check matrix, and

$$G = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{pmatrix}$$

is a generator matrix for the $[4, 2, 3]$ Hamming code over $GF(3)$.

Using SAGE, this can be done easily.

SAGE

```
sage: MS = MatrixSpace(GF(3),2,4)
sage: H = MS([[1,1,1,0],[0,1,2,1]])
sage: C1 = LinearCodeFromCheckMatrix(H); C1
Linear code of length 4, dimension 2 over Finite Field of size 3
sage: G = MS([[1,1,1,0],[1,2,0, 1]])
sage: C2 = LinearCode(H)
sage: C1 == C2
True
```

As you see from the last line, G is indeed a generator matrix for the code defined by the check matrix H.

Definition 15 Let $r > 1$, and let q be a prime power. The *Hamming* $[n, k, 3]$-*code* C over $\mathbb{F}$ is the linear code with

$$n = \left(q^r - 1\right)/(q-1), \qquad k = n - r,$$

and parity check matrix H is defined to be the matrix whose columns are all the (distinct) nonzero vectors in $\mathbb{F}^r$, normalized to have first nonzero coordinate equal to 1.

Remark 2 To be precise, this definition depends on the ordering (1.4.2) of S selected to construct the rows of the check matrix H. Of course, two different orderings of S do not necessarily yield the same code. Nonetheless, the two corresponding codes are both still called Hamming codes.

Let

$$e_1 = (1, 0, \ldots, 0), \quad e_2 = (0, 1, 0, \ldots, 0), \quad \ldots, \quad e_n = (0, \ldots, 0, 1)$$

be the standard basis vectors in $\mathbb{F}^n$.

Here is an algorithm implementing a decoding strategy:

1. Input: A received vector $v \in \mathbb{F}^n$. Assume that v has ≤ 1 error.
   Output: A codeword $c \in C$ closest to v.
2. Compute $H \cdot v$ (regarding v as a column vector). This is an r-tuple, so it must be of the form $a \cdot s$ for some s in the set of "scaled vectors" S constructed above and for some $a \in \mathbb{F}^\times$.
3. If s is the ith element of S (i.e., the ith column of H), then set $c = v - a \cdot e_i$.
4. Return c.

The Hamming codes give, in some sense, the best 1-error-correcting codes (see Niven [Ni]). In particular, if $n = (q^r - 1)/(q-1)$, then there is no shorter 1-error-correcting code of dimension $k = n - r$. However, for $e > 1$, the best e-error-correcting codes of length n is unknown for large n, provided that we assume that e is fixed,[6] e.g., $e = 2$. The search for the best 2-error correcting codes lead to the discovery of the BCH codes around 1960 (see, for example, Hill [Hil]). However, the BCH codes are not known, in general, to provide the best 2-error correcting codes.

Open Problem 1 Find the best linear 2-error-correcting code of length n.

This is also related to "Ulam's game" or "searching with lies," which have an extensive literature. See, for example, the two sections on searching with lies in [JKTu].

[6] If $e > 1$ is allowed to vary with n, then more can be said, but we omit that case.

1.5 Bounds on the Parameters of a Code

In this section, we prove the Singleton bound and the Gilbert–Varshamov bound. For a fixed length n, the Singleton bound is an upper bound on the parameters k, d (the proof given below works for nonlinear codes as well). The Gilbert–Varshamov bound is a lower bound, telling us that there must exist a code with "good parameters." Whereas it is often times very easy to explicitly construct a linear code satisfying this *upper* bound, it is far from easy to find a code satisfying this *lower* bound.

Theorem 16 (The Singleton bound) *Every linear* (n, M, d)*-code* C *over* $\mathbb{F} = GF(q)$ *satisfies*

$$M \leq q^{n-d+1}.$$

Proof Fix a basis of $\mathbb{F}^n$ and write all the codewords in this basis. Delete the first $d-1$ coordinates in each codeword. Call this new code $C' \subset \mathbb{F}^{n-d+1} = V$. Since C has minimum distance d, these codewords of C' are still distinct. There are therefore M of them. But there cannot be more than $q^{n-(d-1)} = q^{n-d+1} = |\mathbb{F}^{n-d+1}|$ of them, since that is the total number of vectors in the remaining vector space V. This gives the inequality. □

A linear code C whose parameters satisfy $k + d = n + 1$ is called *maximum distance separable* or *MDS*. Such codes, when they exist, are in some sense best possible.

Although the classification of MDS codes is not complete (at the time of this writing), there is a great deal known; see, for example, Hirschfeld [Hi]. The following question is still, at the time of this writing, open. A generalized Reed–Solomon code (defined later in this book) is a typical example of an MDS code.

The following conjecture has been open for some time. Very recently, Simeon Ball has announced a solution in the case where q is a prime [Bal].

Open Problem 2 (Main conjecture on MDS codes) For every $[n, k, d]$-code over $GF(q)$ which is MDS, the following is true. If $1 < k < q$, then $n \leq q + 1$, except when q is even and $k = 3$ or $k = q - 1$, in which case $n \leq q + 2$.

Before going on to an example, let us discuss a simple consequence of this. Let C_i be a sequence of linear codes with parameters $[n_i, k_i, d_i]$ such that $n_i \to \infty$ as i goes to infinity and such that both the limits

$$R = \lim_{i\to\infty} \frac{k_i}{n_i}, \qquad \delta = \lim_{i\to\infty} \frac{d_i}{n_i}$$

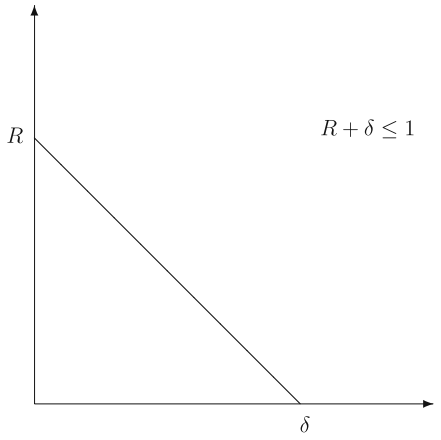
exist. The Singleton bound implies

$$\frac{k_i}{n_i} + \frac{d_i}{n_i} \leq 1 + \frac{1}{n_i}.$$

Letting i tend to infinity, we obtain

$$R + \delta \leq 1.$$

This bound implies that any sequence of codes must have, in the limit, information rate and relative minimum distance in the triangle pictured in the figure below.



Theorem 17 (Gilbert–Varshamov bound) *There exists a code $C \subset \mathbb{F}^n$ such that*

$$\frac{q^n}{\sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i} \leq |C|.$$

Proof Suppose that C has minimum distance d and block length n. Suppose moreover that C is as large as possible with these properties. In other words, we cannot increase the size of C by adding another vector into C and still keeping the minimum distance d. This implies that each $v \in \mathbb{F}^n$ has distance $\leq d-1$ from some codeword in C. This implies $\mathbb{F}^n \subset \bigcup_{c \in C} B(c, d-1, \mathbb{F}^n)$, but since the opposite inclusion is obvious, we have

$$\mathbb{F}^n = \bigcup_{c \in C} B\big(c, d-1, \mathbb{F}^n\big).$$

By Lemma 8, all these balls $B(c, d-1, \mathbb{F}^n)$ have the same cardinality, so if we fix a $c_0 \in C$, then for each $c \in C$, $|B(c, d-1, \mathbb{F}^n)| = |B(c_0, d-1, \mathbb{F}^n)|$. Therefore,

$$\begin{aligned} q^n = \big|\mathbb{F}^n\big| &= \left| \bigcup_{c \in C} B\big(c, d-1, \mathbb{F}^n\big) \right| \\ &\leq \sum_{c \in C} \big| B\big(c, d-1, \mathbb{F}^n\big) \big| \\ &= |C| \cdot \big| B\big(c_0, d-1, \mathbb{F}^n\big) \big| = |C| \sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i. \end{aligned}$$

□

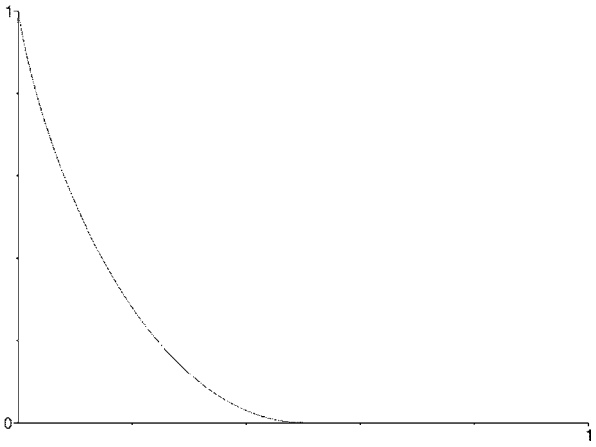


Fig. 1.8 The Gilbert–Varshamov bound for binary codes

Take δ fixed and take $d = [\delta n]$ in the left-hand side of the inequality, call it $M_{n,\delta}$, displayed in Theorem 17. In the (δ, R)-plane, plot δ against $R = \lim_{n\to\infty} \log_q(M_{n,\delta})/n$ as n goes to infinity to obtain Fig. 1.8.

Theorem 18 (Hamming or sphere-packing bound) *For any* (*not necessarily linear*) *code* $C \subset \mathbb{F}^n$ *having* M *elements, we have*

$$M \sum_{i=0}^{e} \binom{n}{i} (q-1)^i \leq q^n,$$

where $e = [(d-1)/2]$.

Proof For each codeword $c \in C$, construct a ball B_c of radius $e = [(d-1)/2]$ about it. These are nonintersecting, by definition of d and the Hamming metric. By Lemma 8, each such ball has

$$\sum_{i=0}^{e} \binom{n}{i} (q-1)^i$$

elements, and there are M such balls. The result follows from the fact that $\bigcup_{c\in C} B_c \subset \mathbb{F}^n$ and $|\mathbb{F}^n| = q^n$. □

A code which attains the equality in the Hamming bound is called *perfect*. For example, the Hamming $[7, 4, 3]$ code is perfect.

1.5.1 Question: What Is "The Best" Code?

What is the "best" code of a given length? This natural, but very hard, question motivates the search for asymptotic bounds discussed in the following subsection.

Asymptotic Bounds—Manin's Theorem

Let

$$R = R(C) = \frac{k}{n},$$

which measures the information rate of the code, and

$$\delta = \delta(C) = \frac{d}{n},$$

which measures the error correcting ability of the code. Let Σ_q denote the set of all $(\delta, R) \in [0,1]^2$ such that there exists a sequence C_i, $i = 1, 2, \ldots$, of $[n_i, k_i, d_i]$-codes for which $\lim_{i\to\infty} d_i/n_i = \delta$ and $\lim_{i\to\infty} k_i/n_i = R$.

The following theorem describes information-theoretical limits on how "good" a linear code can be.

Theorem 19 (Manin) *There exists a continuous decreasing function*

$$\alpha_q : [0,1] \to [0,1],$$

such that

- α_q *is strictly decreasing on* $[0, \frac{q-1}{q}]$,
- $\alpha_q(0) = 1$,
- *if* $\frac{q-1}{q} \le x \le 1$, *then* $\alpha_q(x) = 0$,
- $\Sigma_q = \{(\delta, R) \in [0,1]^2 \mid 0 \le R \le \alpha_q(\delta)\}$.

For more details, see, for example, [SS], Chap. 1.

Open Problem 3 Not a single value of $\alpha_q(x)$ is known for $0 < x < \frac{q-1}{q}$! Can you find one?

For the case $q = 2$, see Conjecture 22 below.

Asymptotic Bounds—The Plotkin Curve

As mentioned above, the value of the bound $\alpha_q(\delta)$ is a mystery. However, we do have the following well-known upper bound.

Theorem 20 (Plotkin bound)

(a) *Suppose* $q > 2$. *If* C *is an* $[n, k, d]_q$*-code and* $d > n(1 - 1/q)$, *then*

$$|C| \le \frac{qd}{qd - (q-1)n}.$$

(b) *Suppose $q = 2$. If C is an $[n, k, d]_2$-code, then*

$$|C| \leq \begin{cases} 2\left[\frac{d}{2d-n}\right], & \text{if } d \leq n \leq 2d, \\ 4d, & \text{if } n = 2d, \end{cases}$$

where $\lfloor \ldots \rfloor$ denotes the floor function.

Levenshtein proved that if Hadamard's conjecture is true, then the binary case of Plotkin's bound is sharp, provided that we allow C to be nonlinear (see p. 333 in Huffman and Pless [HP1], Chap. 9 in Bierbrauer [Bi], and de Launey and Gordon [dLG] for further references and details).

Taking $R = \delta n$ and $d = \delta n$ and letting $n \to \infty$, one can determine an asymptotic form of this bound:

$$\alpha_q(\delta) \leq 1 - \delta/\left(1 - q^{-1}\right).$$

The line $(\delta, 1 - \delta/(1 - q^{-1}), 0 < \delta < 1 - 1/q$, is called the *Plotkin curve*.

Asymptotic Bounds—The Gilbert–Varshamov Curve

It is not known whether or not $\alpha_q(x)$ is differentiable for $0 < x < \frac{q-1}{q}$, nor is it known if $\alpha_q(x)$ is convex on $0 < x < \frac{q-1}{q}$. However, the following estimate is well known.

Theorem 21 (Gilbert–Varshamov) *We have*

$$\alpha_q(x) \geq 1 - x \log_q(q-1) + x \log_q(x) + (1-x) \log_q(1-x).$$

In other words, for each fixed $\epsilon > 0$, there exists an $[n, k, d]$-code C (which may depend on ϵ) with

$$R(C) + \delta(C) \geq 1 - \delta(C) \log_q\left(\frac{q-1}{q}\right) + \delta(C) \log_q\left(\delta(C)\right) + \left(1 - \delta(C)\right) \log_q\left(1 - \delta(C)\right) - \epsilon.$$

The curve $(\delta, 1 - \delta \log_q(\frac{q-1}{q}) + \delta \log_q(\delta) + (1-\delta) \log_q(1-\delta))), 0 < \delta < 1 - 1/q$, is called the *Gilbert–Varshamov curve.*

The plot of the Gilbert–Varshamov curve with several other asymptotic bounds is given in Fig. 1.9.

Here are the SAGE commands which produced Fig. 1.9.

SAGE

```
sage: P1 = plot(gv_bound_asymp(x, 2), x, 0, 1/2, linestyle="dotted")
sage: P2 = plot(plotkin_bound_asymp(x, 2), x, 0.0, 0.50, linestyle="dashed")
sage: P3 = plot(hamming_bound_asymp(x, 2), x, 0.0, 0.50)
sage: show(P1+P2+P3,dpi=300)
```

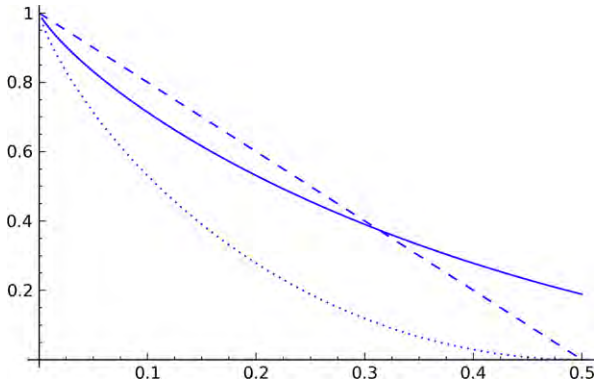


Fig. 1.9 Plot of the Gilbert–Varshamov (*dotted*), Plotkin (*dashed*), and Hamming (*solid*) curves using SAGE

The most that is "known" about the best possible lower bound is a conjecture in the binary case and is stated as follows.

Conjecture 22 *The binary version of the Gilbert–Varshamov bound is asymptotically exact.*

This folklore conjecture was known to the coding theory community since the 1960s. Goppa thought he had a proof at one point so some call this statement "Goppa's conjecture." See Jiang and Vardy [JV] and Gaborit and Zemor [GZ] for recent discussions. Also, it is discussed in much more detail in Chap. 4 below.

1.5.2 The Fake Singleton Bound

Here is an amusing example from Amin Shokrollahi [S].

This example will construct, in a naive way, a set of 2^m vectors, call it C, in $\mathbb{F}_2^{2m}$, and an addition & in C such that C is a binary vector space under & and such that the weight of $c\&c'$ is always at least m (this construction gets you all the way to the "fake Singleton" bound). Here it is.

Let ℓ denote the m-dimensional vector in which all entries are 1. Let $C = \{(a, a+\ell) | a \in \mathbb{F}_2^m\}$. For $c = (a, a+\ell)$ and $c' = (a', a'+\ell)$, let $c\&c' = (a+a', a+a'+\ell)$. Under this "addition," C becomes a binary vector space. The weight of any element in C is m, so the &-distance of any two elements in C, defined as the weight of their &-addition, is m. This is a code of length $n = 2m$, dimension $k = m$, and minimum &-distance $d_\& = m$, so the fake singleton bound $k + d_\& \leq n + 1$ is almost attained. In particular, we have asymptotic rate $R = \frac{1}{2}$ and relatively minimum $\oplus$-distance $\delta_\& = \frac{1}{2}$.

Moral of the story: If you change the basis of your code, do not use the "old" Hamming metric, since it depends on the basis chosen.

1.6 Quadratic Residue Codes and Other Group Codes

In this section, we give examples of several group-theoretical constructions which lead to codes having lots of extra symmetry.

1.6.1 Automorphism Groups

Let $C \subset \mathbb{F}^n$ be a code. An element $g \in S_n$ of the symmetric group of degree n is called a *permutation of* C if the action of g on $\mathbb{F}^n$ restricts to an action on C, i.e., if $gC \subset C$. Define the *permutation automorphism group* of C to be the group of all such permutations. For brevity, we call this group the *automorphism group of* C, denoted $\mathrm{Aut}(C)$. Note that this definition makes sense even if C is a nonlinear code.

1.6.2 Cyclic Codes

Let G denote a cyclic group of order n with generator σ. Suppose that G acts on the set $\{0, 1, \ldots, n-1\}$ by $\sigma(i) = i + 1 \bmod n$. Consider a finite field $\mathbb{F}$ and let us identify σ with the cyclic shift sending $\sigma : \mathbb{F}^n \to \mathbb{F}^n$ sending $(a_0, a_1, \ldots, a_{n-1}) \longmapsto (a_{n-1}, a_0, \ldots, a_{n-2})$, and let $G = \langle \sigma \rangle$.

Definition 23 A linear code C of length n is a *cyclic code* if whenever $c = (c_0, \ldots, c_{n-1})$ is a codeword in C, then so is its cyclic shift $c' = (c_{n-1}, c_0, \ldots, c_{n-2})$.

Example 24 Consider the binary code C with generator matrix

$$G = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}. \tag{1.6.1}$$

Clearly these four rows g_1, g_2, g_3, g_4 are obtained from the previous by a shift to the right. Also note that the shift of g_4 to the right is equal to $g_5 = g_1 + g_3 + g_4$; the shift of g_5 to the right is $g_6 = g_1 + g_2 + g_3$; and the shift of g_6 is $g_7 = g_2 + g_3 + g_4$; the shift of g_7 is g_1. Therefore, the linear code generated by G is invariant under shifts to the right. Therefore, C is a cyclic code.

SAGE

```
sage: MS = MatrixSpace( GF(2), 4, 7)
sage: MS = MatrixSpace(GF(2),4,7)
sage: G = MS([[1,0,1,1,0,0,0],[0,1,0,1,1,0,0],
              [0,0,1,0,1,1,0],[0,0,0,1,0,1,1]])
sage: C = LinearCode(G)
```

```
sage: A = C.automorphism_group_binary_code()
sage: C7 = CyclicPermutationGroup(7)
sage: C7.is_subgroup(A)
True
```

The last command tells us that the automorphism group of the code C contains the groups of cyclic permutations of order 7. This is equivalent to saying that C is cyclic.

Cyclic codewords are conveniently represented as polynomials modulo $x^n - 1$. In fact, if $c = (c_0, \ldots, c_{n-1})$, then let (with apologies for the abuse of notation)

$$c(x) = c_0 + c_1 x + \cdots + c_{n-1} x^{n-1}$$

denote the associated *codeword polynomial*. In this notation the cyclic shift $c' = (c_{n-1}, c_0, \ldots, c_{n-2})$ of c corresponds to the polynomial $xc(x) \pmod{x^n - 1}$. In other words, cyclic shifts correspond to multiplication by x. Since cyclic shifts leave cyclic codes invariant, multiplication by any power of x modulo $x^n - 1$ corresponds to a codeword in C. Since C is a linear code, the sum of any two such codeword polynomials is another codeword polynomial. Therefore, in fact, the product of any codeword polynomial times any polynomial in x modulo $x^n - 1$ is another codeword polynomial.

Denote by R_n the ring of polynomials with coefficients in $\mathbb{F}$ modulo $x^n - 1$:

$$R_n = \mathbb{F}[x]/\big(x^n - 1\big). \tag{1.6.2}$$

Define an *ideal* I of R_n to be any subset of R_n closed under addition and multiplication by an arbitrary element of R_n:

- If $f, g \in I$, then $f + g \in I$, and
- If $f \in I$ and $r \in R_n$ then $rf \in I$.

In other words, an ideal in R_n is simply a subset closed under addition and multiplication by an arbitrary polynomial modulo $x^n - 1$. In particular, the collection of codeword polynomials associated to a cyclic code is an ideal of R_n.

Lemma 25 *There is a natural one-to-one correspondence between cyclic codes of length n over $\mathbb{F}$ and ideals of R_n.*

This can be found in any book on coding theory, for example, MacWilliams and Sloane [MS].

In order to define the generator polynomial of a cyclic code, we need the following mathematical fact.

Lemma 26 *Every ideal I of R_n is of the form $g(x)R_n$. In other words, every element of I is a multiple of $g(x)$ for some polynomial $g(x)$ in R_n.*

Ideals which are of the form $I = g(x)R_n$ are called *principal ideals* and $g(x)$ is called a *generator* of the ideal I.

Proof Suppose not. Let $s(x)$ be a nonzero element in I of smallest degree. Pick an arbitrary nonzero element $f(x)$ in I. By the division algorithm, we can write $f(x) = q(x)s(x) + r(x)$, where q and r are polynomials, and the degree of $r(x)$ is strictly less than the degree of $s(x)$. Notice that $r(x) = f(x) - q(x)s(x)$ belongs to I by definition. This contradicts the assumption that $s(x)$ has smallest degree unless $r(x) = 0$. Therefore, every element of I is a multiple of $s(x)$. Take $g(x) = s(x)$. □

Definition 27 Let C be a cyclic code of length n. Let I be the ideal corresponding to C by Lemma 25. We call $g(x)$ a *generator polynomial* of C if it is a generator of I.

Example 28 We continue with Example 24. Let $g(x) = 1 + x^2 + x^3$. This is the codeword polynomial associated to the top row of the generator matrix in (1.6.1). This polynomial $g(x)$ is the generator polynomial of the cyclic code C in that example. Note that $x^7 - 1 = (x + 1)(x^3 + x^2 + 1)(x^3 + x + 1)$.

We use SAGE to illustrate these claims.

SAGE

```
sage: P.<x> = PolynomialRing(GF(2),"x")
sage: g = x^3+x^2+1
sage: C = CyclicCodeFromGeneratingPolynomial(7,g); C
Linear code of length 7, dimension 4 over Finite Field of size 2
sage: C.gen_mat()
[1 0 1 1 0 0 0]
[0 1 0 1 1 0 0]
[0 0 1 0 1 1 0]
[0 0 0 1 0 1 1]
sage: factor(x^7-1)
(x + 1) * (x^3 + x + 1) * (x^3 + x^2 + 1)
```

1.6.3 Quadratic Residue Codes

Usually quadratic residue codes are constructed as a special type of cyclic code. However, here we define them using Fourier transforms. (For the usual definition, see, for example, [MS].)

Fourier Transforms on Finite Fields

There is a finite field analog of the usual Fourier transform

$$f(x) \longmapsto \int_{\mathbb{R}} f(x)e^{ixy}\,dx$$

on the additive group of the field of real numbers $\mathbb{R}$. (It is doubtful that Fourier had finite fields in mind in the early 1800s when he used Fourier series to solve the heat equation!) First, for the finite-field analog of e^{ixy}, we need to know how to construct the additive characters of $\mathbb{F}$. Recall that an *additive character* of $\mathbb{F}$ is a function $\psi : \mathbb{F} \to \mathbb{C}$ for which $\psi(x+y) = \psi(x)\psi(y)$.

Let $p > 2$ denote an odd prime, and let $(\frac{a}{p})$ denote the *Legendre character*:

$$\left(\frac{a}{p}\right) = \begin{cases} 1, & a \neq 0 \text{ quadratic residue mod } p, \\ -1, & a \neq 0 \text{ quadratic nonresidue mod } p, \\ 0, & a = 0, \end{cases}$$

for $a \in GF(p)$. By *quadratic reciprocity*, if $p > 2$, we have $(\frac{2}{p}) = (-1)^{\frac{p^2-1}{8}}$. If p, ℓ are both odd primes, then we have $(\frac{\ell}{p})(\frac{p}{\ell}) = (-1)^{\frac{(p-1)(\ell-1)}{4}}$. In particular, 2 is a quadratic residue of p if and only if $p \equiv \pm 1 \pmod 8$.

Let $\mathbb{F} = GF(p)$, and let $F = GF(\ell)$, where ℓ is a prime different from p which is a quadratic residue of p. For example, we shall take $\ell = 2$ and $p \equiv 1 \pmod 8$. Let ξ be a nontrivial pth root of unity in a field containing F.

Define $\psi_1 : \mathbb{F} \to F(\xi)^\times$ by $\psi_1(a) = \xi^a$, $a \in \mathbb{F}$. Clearly, $\psi_1(a_1 + a_2) = \xi^{a_1+a_2} = \xi^{a_1}\xi^{a_2} = \psi_1(a_1)\psi_1(a_2)$ for all $a_1, a_2 \in \mathbb{F}$, so ψ_1 is an additive character. For any $b \in \mathbb{F}$, define

$$\psi_b(a) = \psi_1(ab).$$

In particular, $\psi_0 = 1$. Since $\psi_b(a_1 + a_2) = \psi_b(a_1)\psi_b(a_2)$ for all $a_1, a_2 \in \mathbb{F}$, it follows that ψ_b too is an additive character.

Definition 29 Let $f : \mathbb{F} \to F(\xi)$ be any function. The *Fourier transform* of f is the function

$$FT_f(b) = \sum_{a \in \mathbb{F}} f(a)\psi_b(a), \quad b \in \mathbb{F}.$$

The following property of additive characters is needed to prove basic facts about the Fourier transform needed later.

Lemma 30 (a) (Orthogonality) *As elements of F, we have*

$$\sum_{c \in \mathbb{F}} \psi_a(c)\psi_b(c) = \begin{cases} p, & a + b = 0, \\ 0, & a + b \neq 0. \end{cases}$$

(*Note*: *if $\ell = 2$, then here $p = 1$ in F.*)

(b) *If $\psi : \mathbb{F} \to F(\xi)$ is any additive character of $\mathbb{F}$, then there is a unique $b \in \mathbb{F}$ such that $\psi = \psi_b$.*

The first part is a special case of "Schur orthogonality." The second part is a special case of the duality between elements of an Abelian group and its dual group of characters. A proof can be found in many books on group theory or finite fields.

Lemma 31 (Fourier inversion) *If $f : \mathbb{F} \to F(\xi)$ is any function,*

$$f(a) = |\mathbb{F}|^{-1} \sum_{b \in \mathbb{F}} FT_f(b)\psi_b(-a), \quad a \in \mathbb{F}.$$

(*Recall that $|\mathbb{F}|^{-1}$ is to be regarded as an element of $F(\xi)$.*)

This is a consequence of orthogonality.

Generalized Quadratic Residue Codes

If "useful and practical" fought "mathematically beautiful" in a battle over the quadratic residue (QR) codes, then probably "mathematically beautiful" would win. These QR codes seems to have reasonably fast encoders and decoders but lack good parameters.[7] However, these QR codes have striking mathematical properties, especially as related to representation theory, as we shall see.[8]

Again, let ℓ, p be primes with $p > 2$, and $\ell \geq 2$ a quadratic residue of p.

Let Q denote the set of quadratic residues in $\mathbb{F}^\times$, and N denote the set of non-quadratic residues in $\mathbb{F}^\times$. In other words, $a \in Q$ if and only if $(\frac{a}{p}) = 1$, and $a \in N$ if and only if $(\frac{a}{p}) = -1$. Since $(\frac{*}{p})$ defines a nontrivial character of $\mathbb{F}^\times$, orthogonality implies $\sum_{a \in \mathbb{F}^\times} (\frac{a}{p}) = 0$. This implies $|Q| = |N|$, so $|Q| = \frac{1}{2}|\mathbb{F}^\times| = |N|$.

Let us enumerate the elements of $\mathbb{F} = GF(p)$ in some way, say $\mathbb{F} = \{0, 1, \ldots, p-1\}$. Now identify $GF(\ell)^p$ with the vector space of function values

$$\big\{\big(f(0), f(1), \ldots, f(p-1)\big) \mid f : \mathbb{F} \to GF(\ell)\big\}.$$

The *generalized quadratic residue (GRS) code* is the subspace of functions in the kernel of the Fourier transform on Q:

$$C_Q(\mathbb{F}, F) = \big\{\big(f(0), f(1), \ldots, f(p-1)\big) \,|\, FT_f(a) = 0 \ \forall a \in Q\big\}.$$

There is an analogous code for the nonresidues:

$$C_N(\mathbb{F}, F) = \big\{\big(f(0), f(1), \ldots, f(p-1)\big) \,|\, FT_f(a) = 0 \ \forall a \in N\big\}.$$

Though tempting, this last one is not called the generalized quadratic nonresidue code! Instead, usually these two are simply referred to as the generalized quadratic residue codes.

---

[7] Although, please see Chap. 4, where some interesting but conjectural results use quadratic residue code-like constructions to find related codes which might have very good parameters.

[8] We follow [MS], Sects. 16.4–16.5.

Let $Q = \{a_1, \ldots, a_r\}$ (so $r = \frac{p-1}{2}$). From this definition we see that a check matrix for $C_Q(\mathbb{F}, F)$ is

$$H = \begin{pmatrix} \psi_{a_1}(0) & \psi_{a_1}(1) & \ldots & \psi_{a_1}(p-1) \\ \psi_{a_2}(0) & \psi_{a_2}(1) & \ldots & \psi_{a_2}(p-1) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_{a_r}(0) & \psi_{a_r}(1) & \ldots & \psi_{a_r}(p-1) \end{pmatrix} = \begin{pmatrix} 1 & \xi^{a_1} & \ldots & \xi^{a_1(p-1)} \\ 1 & \xi^{a_2} & \ldots & \xi^{a_2(p-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \xi^{a_r} & \ldots & \xi^{a_r(p-1)} \end{pmatrix}.$$

Lemma 32 *The parameters* $[n, k, d]$ *of the generalized quadratic residue codes satisfy*

$$n = p, \qquad k = \frac{p+1}{2}, \qquad d \geq \sqrt{p}.$$

Determining d for quadratic residue codes with p "large" is an open problem in general. For example (as of this writing, recently), M. Grassl has published some tables of values $[n, k, d]$ for quadratic residue codes with $\ell = 2, 3$ and $p \leq 167$ (see also Voloch [V3] and Chap. 16 of [MS]).

Open Problem 4 Determine which coordinates the information bits lie in for the family of quadratic residue codes.

Let $\overline{C}_Q(\mathbb{F}, F)$ denote the code generated by $C_Q(\mathbb{F}, F)$ and the all-1s vector, and $\overline{C}_N(\mathbb{F}, F)$ denote the code generated by $C_N(\mathbb{F}, F)$ and the all-1s vector.

Lemma 33 *We have*

$$C_Q(\mathbb{F}, F)^{\perp} = \begin{cases} \overline{C}_Q(\mathbb{F}, F), & p \equiv 1 \pmod 4, \\ \overline{C}_N(\mathbb{F}, F), & p \equiv -1 \pmod 4, \end{cases}$$

and

$$C_N(\mathbb{F}, F)^{\perp} = \begin{cases} \overline{C}_N(\mathbb{F}, F), & p \equiv 1 \pmod 4, \\ \overline{C}_Q(\mathbb{F}, F), & p \equiv -1 \pmod 4, \end{cases}$$

(This is proven in Sect. 16.4 in [MS].) In other words, if $p \equiv 1 \pmod 4$, then all the codewords in the code $C = C_Q(\mathbb{F}, F)$ are orthogonal to each other! (Such a code is sometimes called *self-orthogonal*. If, in addition, every vector orthogonal to all the codewords is a codeword itself, then the code is called *self-dual*.)

Extended Quadratic Residue Codes

Define the *extended quadratic residue codes* by

$$\hat{C}_Q(\mathbb{F}, F) = \left\{ (c_1, \ldots, c_p, c_\infty) \mid (c_1, \ldots, c_p) \in C_Q(\mathbb{F}, F),\ c_\infty = \alpha \sum_{i=1}^{p} c_i \right\},$$

$$\hat{C}_N(\mathbb{F}, F) = \left\{ (c_1, \ldots, c_p, c_\infty) \;\middle|\; (c_1, \ldots, c_p) \in C_N(\mathbb{F}, F),\ c_\infty = \alpha \sum_{i=1}^{p} c_i \right\},$$

where $1 + \alpha^2 p = 0$ (either choice of sign will work). These codes are self-dual if $p \equiv 1 \pmod 4$ and are the dual of each other if $p \equiv -1 \pmod 4$.

Even more interesting is the fact that these codes have large automorphism groups.

Theorem 34 (Gleason–Prange) *Assume that* $\ell = 2$ *and* $p \equiv \pm 1 \pmod 8$. *The automorphism group* $\mathrm{Aut}(\hat{C}_Q(\mathbb{F}, F))$ *contains a subgroup isomorphic to* $PSL(2, p)$.

See [MS], Sect. 16.5 for a proof of this and more details on how the permutation automorphism acts on the code (see also Sect. 6.6 of [HP1]). This theorem says that $\hat{C}_Q(\mathbb{F}, F)$ may be regarded as a representation space of $PSL(2, p)$. The action of $G = PSL(2, p)$ on $C = \hat{C}_Q(\mathbb{F}, F)$ is reminiscent of the Weil representation of $SL(2)$ over a p-adic field, one of the more remarkable representations in mathematics. See Ward [War] for more of this fascinating story.

Chapter 2
Self-dual Codes, Lattices, and Invariant Theory

One of the most interesting fields in all of mathematics concerns the interaction between the fields of integral lattices, modular forms, invariant theory, and error-correcting codes. There are several excellent presentations in the literature of this subject, for example, Conway and Sloane [CS1], Ebeling [Eb], Elkies [E3], Sloane [Sl], and Brualdi, Huffman, and Pless [BHP]. Therefore, this chapter will be brief and refer to these works for details.

Topics treated in this chapter include (a) invariant theory and the relationship with self-dual codes, (b) lattices and connections with binary codes, and (c) optimal, divisible, and extremal codes.

Some open questions which arise are: Which polynomials $F(x, y)$ occur as the weight enumerators of linear codes? Does there exist a binary self-dual [72, 36, 16] code? These questions and others are discussed below.

2.1 Weight Enumerators

The *(Hamming) weight enumerator polynomial* A_C is defined by

$$A_C(x, y) = \sum_{i=0}^{n} A_i x^{n-i} y^i = x^n + A_d x^{n-d} y^d + \cdots + A_n y^n,$$

where

$$A_i = \left|\{c \in C \mid \mathrm{wt}(c) = i\}\right|$$

denotes the number of codewords of weight i. The *support of* C is the set $\mathrm{supp}(C) = \{i \mid A_i \neq 0\}$. If $A_C(x, y) = A_{C^\perp}(x, y)$, then C is called a *formally self-dual code*. The *spectrum of* C is the list of coefficients of A_C:

$$\mathrm{spec}(C) = [A_0, \ldots, A_n].$$

We say that two codes are *formally equivalent* if they have the same spectrum.

D. Joyner, J.-L. Kim, *Selected Unsolved Problems in Coding Theory*,
Applied and Numerical Harmonic Analysis,
DOI 10.1007/978-0-8176-8256-9_2,

Example 35 Let

$$G = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix},$$

and let C be the binary code generated by G. This code has the spectrum

$$\operatorname{spec}(C) = [1, 0, 0, 0, 15, 0, 15, 0, 0, 0, 1],$$

and so does its dual code $C^{\perp}$, although $C \neq C^{\perp}$. So, this code is formally self-dual but not self-dual.

We say that two codes C, C' are *isometric* if there is a bijective linear transformation $\phi : C \to C'$ between them that preserves the Hamming distance function: for all $c_1, c_2 \in C$, we have

$$d(c_1, c_2) = d\big(\phi(c_1), \phi(c_2)\big).$$

Saying that two codes are formally equivalent is weaker than saying that the codes are isometric. In fact, it is known that two codes are monomially equivalent if and only if they are isometric (by a result of MacWilliams). In particular, two binary codes are (permutation) equivalent if and only if they are isometric.

Example 36 Indeed, there exist many examples of inequivalent binary codes C, C' which are formally equivalent. Here is an example using SAGE.

SAGE

```
sage: G1 = matrix(GF(2),[[1,1,0,0,0,0],[0,0,1,1,0,0],[0,0,0,0,1,1]])
sage: G2 = matrix(GF(2),[[1,1,0,0,0,0],[1,0,1,0,0,0],[1,1,1,1,1,1]])
sage: C1 = LinearCode(G1)
sage: C2 = LinearCode(G2)
sage: C1.is_permutation_equivalent(C2)
False
sage: C1.weight_distribution()
[1, 0, 3, 0, 3, 0, 1]
sage: C2.weight_distribution()
[1, 0, 3, 0, 3, 0, 1]
```

In other words, the binary codes C_1, C_2 having the generator matrices

$$G_1 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}, \qquad G_2 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix},$$

respectively, are formally equivalent but inequivalent.

Open Problem 5 Given a homogeneous polynomial

$$F(x, y) = x^n + \sum_{i=1}^{n} f_i x^{n-i} y^i$$

of degree n with nonnegative integer coefficients $f_i \in \mathbb{Z}$, find necessary and sufficient conditions (short of enumerating all weight enumerators of linear codes with length n) which determine whether or not $F(x, y) = A_C(x, y)$ for some linear code C of length n.

Index the finite field $GF(q)$ in some fixed way: $GF(q) = \{\omega_0, \omega_1, \ldots, \omega_{q-1}\}$ with $\omega_0 = 0$. The *composition* of $v = (v_1, \ldots, v_n) \in GF(q)^n$ is defined by

$$\mathrm{comp}(v) = (s_0, \ldots, s_{q-1}),$$

where $s_j = s_j(v)$ denotes the number of components of v equal to ω_j. Clearly, $\sum_{i=0}^{q-1} s_i(v) = n$ for each $v \in GF(q)^n$. For $s = (s_0, \ldots, s_{q-1}) \in \mathbb{Z}^q$, let $T_C(s)$ denote the number of codewords $c \in C$ with $\mathrm{comp}(c) = s$. Define the *complete weight enumerator* by

$$W_C(z_0, \ldots, z_{q-1}) = \sum_{c \in C} z_0^{s_0(c)} \cdots z_{q-1}^{s_{q-1}(c)} = \sum_{s \in \mathbb{Z}^q} T_C(s) z_0^{s_0} \cdots z_{q-1}^{s_{q-1}}. \qquad (2.1.1)$$

Sometimes, when it is convenient, we identify the variables z_i with the variables z_{ω_i}. This enumerator is related to the Hamming weight enumerator as follows:

$$A_C(x, y) = W_C(x, y, \ldots, y).$$

2.2 Divisible Codes

If $b > 1$ is an integer and $\mathrm{supp}(C) \subset b\mathbb{Z}$, then the code C is called *b-divisible*.

Definition 37 Let C be a b-divisible code. If C and $C^\perp$ are both binary and contain the all-ones codeword, then C is said to be *Type 2 divisible*. We say that C is *Type 1 divisible* if C is not of Type 2.

For example, if C is a binary self-dual code, then it must be 2-divisible (since the codeword must be orthogonal to itself, it hence has even weight). This implies that $C^\perp$ contains the all-ones vector. But $C = C^\perp$, so C must be Type 2.

Type 1 codes need not be binary.

Lemma 38 *If C is Type 1 divisible, then*

$$d + bd^\perp \le n + b(b + 1).$$

If C is Type 2 divisible, then

$$2d + bd^{\perp} \leq n + b(b+2).$$

Proof See Theorem 1 in Duursma [D3]. □

As a corollary, we see that if C is formally self-dual, then

$$d \leq \begin{cases} \left[\frac{n}{b+1}\right] + b, & \text{Type 1}, \\ \left[\frac{n}{b+2}\right] + b, & \text{Type 2}. \end{cases}$$

The Gleason–Pierce theorem[1] basically says that, other than a family of uninteresting examples, the formally self-dual divisible codes fall into one of the following four types.

Definition 39 Let C be a formally self-dual b-divisible $[n,k,d]_q$-code. We say that C is *Type I* (*singly even* or *formally self-dual even*) if $q = b = 2$ and n is even. We say that C is *Type II* (or *doubly even*) if $q = 2$, $b = 4$, and $8|n$. We say that C is *Type III* if $q = b = 3$ and $4|n$. If $q = 4$, $b = 2$, and n is even, then C is said to be *Type IV*. In the above definitions, the dual of C is defined with respect to the Euclidean inner product except for the Type IV, which is with respect to the Hermitian inner product.

Lemma 40 (Upper bounds) *Suppose that C is a formally self-dual b-divisible $[n,k,d]_q$-code for $q = 2, 3,$ or 4. Then Type II, Type III, and Type IV codes are all self-dual. Furthermore,*

$$d \leq \begin{cases} 2[n/8] + 2 & \text{if } C \text{ is Type I self-dual}, \\ 4[n/24] + 4 & \text{if } C \text{ is Type II}, \\ 3[n/12] + 3 & \text{if } C \text{ is Type III}, \\ 2[n/6] + 2 & \text{if } C \text{ is Type IV}. \end{cases}$$

Proof This is Theorem 9.3.1 in [HP1]. See also [D3] for a different approach. □

These upper bounds are sometimes referred to as the *Mallows–Sloane bounds*. In fact, the "Type I bound" even holds for formally self-dual even codes (see Theorem 9.3.1 in [HP1], Sect. 11.1 in [NRS]). For a further generalization, see Sect. 7.4.2 below.

A code is called *optimal* if its minimum distance is maximal among all linear codes of that length and dimension. A formally self-dual code C is called *extremal* if the bound in Lemma 40 holds with equality. For example, the parameters $n = 72$, $k = 36$, $d = 16$ meet the conditions of an extremal code of Type II.

[1] See Theorem 2.5.1 in [NRS], also Theorem 89 below.

Below is a table of the parameters of the best possible Type I codes up to length 32 (see, for example, Table 1 in [CS3]):

n	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32
d	2	2	2	2	4	4	4	4	4	6	6	6	6	6	8

Remark 3 It is known that any two extremal codes (if they exist) have the same weight enumerator polynomial (in fact, they are essentially determined in Duursma [D3]).

It is known that there exist only finitely many extremal codes. Zhang [Z] proved the following result (see also [D3] for a short proof, Sect. 11.1 in [NRS], or Huffman and Pless [HP1], p. 345).

Theorem 41 *There is no extremal formally self-dual code with code length n for*:

(i) *binary formally self-dual even code*: $n = 8i$ $(i \geq 4)$, $n = 8i + 2$ $(i \geq 5)$, $8i + 4$ $(i \geq 6)$, $8i + 6$ $(i \geq 7)$;
(ii) *Type II code*: $n = 24i$ $(i \geq 154)$, $24i + 8$ $(i \geq 159)$, $24i + 16$ $(i \geq 164)$;
(iii) *Type III code*: $n = 12i$ $(i \geq 70)$, $12i + 4$ $(i \geq 75)$, $12i + 8$ $(i \geq 78)$;
(iv) *Type IV code*: $n = 6i$ $(i \geq 17)$, $6i + 2$ $(i \geq 20)$, $6i + 4$ $(i \geq 22)$.

This tells us that there are indeed only finitely many extremal codes. However, some gaps remain in our knowledge.

Open Problem 6 Completely classify all extremal codes. In particular, determine whether there exists a binary self-dual $[72, 36, 16]$ code.

For further discussion of the problem, see the webpage [Do]. On that webpage a monetary reward is offered for a solution to this problem. We will discuss more in Sect. 2.6.

In [HK], Han and Kim defined near-extremal formally self-dual codes and proved a bound similar to Theorem 41. Recall that an *additive code over $GF(4)$ of length n* is an additive subgroup of $GF(4)^n$ (we refer to [GHKP] for further details).

Definition 42 A formally self-dual code of length n with minimum distance d below is called *near-extremal*.

(i) binary formally self-dual even code: $d = 2[\frac{n}{8}]$;
(ii) Type II code: $d = 4[\frac{n}{24}]$;
(iii) Type III code: $d = 3[\frac{n}{12}]$;
(iv) Type IV code: $d = 2[\frac{n}{6}]$;
(iv)′ additive self-dual even code over $GF(4)$: $d = 2[\frac{n}{6}]$;
(v) formally self-dual additive odd code over $GF(4)$: $d = [\frac{n}{2}]$.

Theorem 43 *There is no near-extremal code with length n for*

(i) *binary formally self-dual even code*: *if* $n = 8i$ $(i \geq 9)$, $8i + 2$ $(i \geq 12)$, $8i + 4$ $(i \geq 13)$, $8i + 6$ $(i \geq 14)$;
(ii) *Type II code*: *if* $n = 24i$ $(i \geq 315)$, $24i + 8$ $(i \geq 320)$, $24i + 16$ $(i \geq 325)$;
(iii) *Type III code*: *if* $n = 12i$ $(i \geq 147)$, $12i + 4$ $(i \geq 150)$, $12i + 8$ $(i \geq 154)$;
(iv) *Type IV code*: *if* $n = 6i$ $(i \geq 38)$, $6i + 2$ $(i \geq 41)$, $6i + 4$ $(i \geq 43)$;
(iv)′ *additive self-dual even code over* $GF(4)$: *if* $n = 6i$ $(i \geq 38)$, $6i + 2$ $(i \geq 41)$, $6i + 4$ $(i \geq 43)$;
(v) *formally self-dual additive odd code over* $GF(4)$: *if* $n = 2i$ $(i \geq 8)$, $2i + 1$ $(i \geq 10)$.

If $C^\perp$ denotes the dual code of C with parameters $[n, n-k, d^\perp]$, then the *MacWilliams identity*[2] relates the weight enumerator of $C^\perp$ to that of C:

$$A_{C^\perp}(x, y) = |C|^{-1} A_C\big(x + (q-1)y, x - y\big).$$

In particular, C is formally self-dual if and only if $F = A_C$ satisfies the invariance condition

$$F(x, y) = F\left(\frac{x + (q-1)y}{\sqrt{q}}, \frac{x - y}{\sqrt{q}}\right). \tag{2.2.1}$$

Example 44 The following examples are taken from Sloane [Sl]. The notation is as in [Sl] and will be used in the statement of Theorem 45 below.

1. $W_1(x, y) = x^2 + y^2$ is the weight enumerator of the Type I code $C = \{(0, 0), (1, 1)\}$.
2. $W_5(x, y) = x^8 + 14x^4y^4 + y^8$ is the weight enumerator of the Type II $[8, 4, 4]$ code C constructed by extending the binary $[7, 4, 3]$ Hamming code by a check bit. This is the smallest Type II code.
3. $W_6(x, y) = x^{24} + 759x^{16}y^8 + 2576x^{12}y^{12} + 759x^8y^{16} + y^{24}$ is the weight enumerator of the extended binary Golay code with parameters $[24, 12, 8]$.
4. $W_8(x, y) = x^{48} + 17296(x^{36}y^{12} + x^{12}y^{36}) + 535095(x^{32}y^{16} + x^{16}y^{32}) + 3995376(x^{28}y^{20} + x^{20}y^{28}) + 7681680x^{24}y^{24}$ is the weight enumerator of the extended binary quadratic residue code associated to $p = 47$ with parameters $[48, 24, 16]$.
5. $W_9(x, y) = x^4 + 8xy^3$ is the weight enumerator of the Type III ternary code C with generator matrix

$$G = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & -1 & 1 \end{pmatrix}$$

and parameters $[4, 2, 3]$.
6. $W_{10}(x, y) = x^{12} + 264x^6y^6 + 440x^3y^9 + 24y^{12}$ is the weight enumerator of the Type III ternary Golay code with parameters $[12, 6, 6]$.

[2] This is proven in Sect. 7.4.

7. $W_{11}(x, y) = x^2 + 3y^2$ is the weight enumerator of the Type IV code $C = \{(0,0), (1,1), (\alpha,\alpha), (\alpha^2,\alpha^2)\}$, with parameters $[2, 1, 2]$. Here α is a generator of $GF(4)$ satisfying $\alpha^2 + \alpha + 1 = 0$.
8. $W_{12}(x, y) = x^6 + 45x^2y^4 + 18y^6$ is the weight enumerator of the Type IV code C with generator matrix

$$G = \begin{pmatrix} 1 & 0 & 0 & 1 & \alpha & \alpha \\ 0 & 1 & 0 & \alpha & 1 & \alpha \\ 0 & 0 & 1 & \alpha & \alpha & 1 \end{pmatrix}$$

with parameters $[6, 3, 4]$. (Again, α is a generator of $GF(4)$ satisfying $\alpha^2 + \alpha + 1 = 0$.)

2.3 Some Invariants

The following result collects together several facts from Sect. 8.1 in Sloane [Sl].

Recall that divisible codes were defined in Sect. 2.2. These are classified into families called Types in Theorem 89 below. We say that $A_C(x, y)$ is invariant under a linear transformation $T = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ if $A_C(x, y) = A_C(ax + by, cx + dy)$.

Theorem 45 *Assume that C is a formally self-dual divisible code of Type I, II, III, or IV.*

- *I. If C is of Type I, then $A_C(x, y)$ is invariant under the group*

$$G_I = \left\langle \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \right\rangle$$

  of order 16. *Moreover,* $\mathbb{C}[x, y]^{G_I} = \mathbb{C}[W_1, W_5]$.
- *II. If C is of Type II, then $A_C(x, y)$ is invariant under the group*

$$G_{II} = \left\langle \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \right\rangle$$

  of order 192. *Moreover,* $\mathbb{C}[x, y]^{G_{II}} = \mathbb{C}[W_5, W_6]$.
- *III. If C is of Type III, then $A_C(x, y)$ is invariant under the group*

$$G_{III} = \left\langle \sigma, \begin{pmatrix} 1 & 0 \\ 0 & \omega \end{pmatrix} \right\rangle, \qquad \sigma = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & 1 \\ 2 & -1 \end{pmatrix},$$

  of order 48, *where* $\omega \in \mathbb{C} - \{1\}$, $\omega^3 = 1$. *Moreover,* $\mathbb{C}[x, y]^{G_{III}} = \mathbb{C}[W_9, W_{10}]$.
- *IV. If C is of Type IV, then $A_C(x, y)$ is invariant under the group*

$$G_{IV} = \left\langle \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 3 & -1 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \right\rangle$$

  of order 12. *Moreover,* $\mathbb{C}[x, y]^{G_{IV}} = \mathbb{C}[W_{11}, W_{12}]$.

Here are some computations illustrating the above theorem.

Example 46 Here is some SAGE code for computing the invariants of the group G generated by $g_1 = \left(\begin{smallmatrix} 1/\sqrt{q} & 1/\sqrt{q} \\ (q-1)/\sqrt{q} & -1/\sqrt{q} \end{smallmatrix}\right)$ with $q = 2$, $g_2 = \left(\begin{smallmatrix} -1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$, and $g_3 = \left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$. The SAGE method `invariant_generators` calls Singular [GPS], which has methods implemented (due to Simon King) to compute group invariants.

SAGE

```
sage: F = CyclotomicField(8)
sage: z = F.gen()
sage: a = z+1/z
sage: a^2
2
sage: MS = MatrixSpace(F,2,2)
sage: b = -1
sage: g1 = MS([[1/a,1/a],[1/a,-1/a]])
sage: g2 = MS([[b,0],[0,1]])
sage: g3 = MS([[1,0],[0,b]])
sage: G = MatrixGroup([g1,g2,g3])
sage: G.invariant_generators()
[x1^2 + x2^2,
 x1^8 + 28/9*x1^6*x2^2 + 70/9*x1^4*x2^4 + 28/9*x1^2*x2^6 + x2^8]
```

It is not hard to check that this is equivalent with part I of Theorem 45.

Example 47 Here is some SAGE code for computing the invariants of the group G generated by $g_1 = \left(\begin{smallmatrix} 1/\sqrt{q} & (q-1)/\sqrt{q} \\ 1/\sqrt{q} & -1/\sqrt{q} \end{smallmatrix}\right)$ with $q = 2$, $g_2 = \left(\begin{smallmatrix} i & 0 \\ 0 & 1 \end{smallmatrix}\right)$, and $g_3 = \left(\begin{smallmatrix} 1 & 0 \\ 0 & i \end{smallmatrix}\right)$.

SAGE

```
sage: F = CyclotomicField(8)
sage: z = F.gen()
sage: a = z+1/z
sage: b = z^2
sage: MS = MatrixSpace(F,2,2)
sage: g1 = MS([[1/a,1/a],[1/a,-1/a]])
sage: g2 = MS([[b,0],[0,1]])
sage: g3 = MS([[1,0],[0,b]])
sage: G = MatrixGroup([g1,g2,g3])
sage: G.order()
192
sage: G.invariant_generators()
[x1^8 + 14*x1^4*x2^4 + x2^8,
x1^24 + 10626/1025*x1^20*x2^4 + 735471/1025*x1^16*x2^8\
+ 2704156/1025*x1^12*x2^12 + 735471/1025*x1^8*x2^16\
+ 10626/1025*x1^4*x2^20 + x2^24]
```

The above group G leaves invariant the weight enumerator of any self-dual doubly even binary code. The above result implies that any such weight enumer-

ator must be a polynomial in $x^8 + 14x^4y^4 + y^8$ and $1025x^{24} + 10626x^{20}y^4 + 735471x^{16}y^8 + 2704156x^{12}y^{12} + 735471x^8y^{16} + 10626x^4y^{20} + 1025y^{24}$. Using SAGE's Gröbner bases algorithms, it is not hard to check that this is equivalent with part II of Theorem 45. The details are omitted.

Example 48 Here is some SAGE code for computing the invariants of the group G generated by $g_1 = \begin{pmatrix} 1/\sqrt{q} & 1/\sqrt{q} \\ (q-1)/\sqrt{q} & -1/\sqrt{q} \end{pmatrix}$ with $q = 3$, $g_2 = \begin{pmatrix} \omega & 0 \\ 0 & 1 \end{pmatrix}$, and $g_3 = \begin{pmatrix} 1 & 0 \\ 0 & \omega \end{pmatrix}$.

SAGE

```
sage: F = CyclotomicField(12)
sage: z = F.gen()
sage: a = z+1/z
sage: b = z^4
sage: a^2; b^3
3
1
sage: MS = MatrixSpace(F,2,2)
sage: g1 = MS([[1/a,1/a],[2/a,-1/a]])
sage: g2 = MS([[b,0],[0,1]])
sage: g3 = MS([[1,0],[0,b]])
sage: G = MatrixGroup([g1,g2,g3])
sage: G.order()
144
sage: G.invariant_generators()

[x1^12 + (-55/2)*x1^9*x2^3 + 231/16*x1^6*x2^6
 + (-55/128)*x1^3*x2^9 + 61/1024*x2^12,
 x1^12 + 4*x1^9*x2^3 + 21/8*x1^6*x2^6 + 67/64*x1^3*x2^9
 + (-1/512)*x2^12]
```

Example 49 Here is some SAGE code for computing the invariants of the group G generated by $g_1 = \begin{pmatrix} 1/\sqrt{q} & 1/\sqrt{q} \\ (q-1)/\sqrt{q} & -1/\sqrt{q} \end{pmatrix}$ with $q = 4$, $g_2 = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$, and $g_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.

SAGE

```
sage: q = 4; a = 2
sage: MS = MatrixSpace(QQ, 2, 2)
sage: g1 = MS([[1/a,1/a],[(q-1)/a, -1/a]])
sage: g2 = MS([[-1,0],[0,1]])
sage: g3 = MS([[1,0],[0,-1]])
sage: G = MatrixGroup([g1,g2,g3])
sage: G.order()
12
sage: G.invariant_generators()
[x1^2 + 1/3*x2^2,
 x1^6 + 5/3*x1^4*x2^2 + 5/27*x1^2*x2^4 + 11/243*x2^6]
```

Whereas Theorem 45 narrowed down the types of polynomials of two variables which can occur as some $A_C(x, y)$, the following result in some sense computes their "average value," i.e., the so-called *mass formulas*.

Let $\mathrm{Aut}(C)$ denote the permutation automorphism group of C.

Theorem 50 (Theorem 9.1.1, [Sl]) *Assume that C is a formally self-dual divisible code of Type I, II, III, or IV.*

- I. *If $\mathcal{E}$ denotes a complete set of representatives of equivalence classes of Type I codes of length n, then*

$$\sum_{C\in\mathcal{E}} \frac{A_C(x,y)}{|\mathrm{Aut}(C)|} = \frac{1}{n!}\prod_{j=1}^{\frac{n}{2}-2}(2^j+1)\left[2^{\frac{n}{2}-1}(x^n+y^n)+\sum_{2|i}\binom{n}{i}x^{n-1}y^i\right].$$

- II. *If $\mathcal{E}$ denotes a complete set of representatives of equivalence classes of Type II codes of length n, then*

$$\sum_{C\in\mathcal{E}} \frac{A_C(x,y)}{|\mathrm{Aut}(C)|} = \frac{1}{n!}\prod_{j=1}^{\frac{n}{2}-3}(2^j+1)\left[2^{\frac{n}{2}-2}(x^n+y^n)+\sum_{4|i}\binom{n}{i}x^{n-1}y^i\right].$$

- III. *If $\mathcal{E}$ denotes a complete set of representatives of equivalence classes of Type III codes of length n, then*

$$\sum_{C\in\mathcal{E}} \frac{CWE_C(x,y,z)}{|\mathrm{Aut}(C)|} = \frac{1}{2^{n-1}n!}\prod_{j=1}^{\frac{n}{2}-2}(3^j+1)\left[3^{\frac{n}{2}-1}x^n+\sum_{3|i}2^i\binom{n}{i}x^{n-1}y^i\right].$$

- IV. *If $\mathcal{E}$ denotes a complete set of representatives of equivalence classes of Type IV codes of length n, then*

$$\sum_{C\in\mathcal{E}} \frac{A_C(x,y)}{|\mathrm{Aut}(C)|} = \frac{1}{3^n n!}\prod_{j=1}^{\frac{n}{2}-2}(2^{2j+1}+1)\left[2^{n-1}x^n+\sum_{3|i}3^i\binom{n}{i}x^{n-1}y^i\right].$$

2.4 Codes over Other Finite Rings

More generally, let R be a finite commutative ring with identity. A *linear code C of length n over R* is an R-submodule of R^n (with a fixed basis). In other words, we insist that C be closed under addition and R-scalar multiples. Codes over a ring of integers modulo m and more exotic ring structures (e.g., Galois rings, chain rings, principal ideal rings, and Frobenius rings) have been actively studied by many authors.

In what follows, we pay special attention to the case where $R = \mathbb{F}_q = GF(q)$, $\mathbb{Z}/2k\mathbb{Z}$, or $\mathbb{Z}/m\mathbb{Z}$. There is an interesting connection with lattices in these cases. Let $|\ldots|$ denote an injection

$$|\ldots| : R \to \mathbb{Z}_{>0}.$$

We collect facts from [BDHO].

Let C be a linear code over R of length n, and $G(C)$ be a generator matrix whose rows generate C. Let $x = (x_1, \ldots, x_n) \in R^n$. The *Hamming weight* $\mathrm{wt}_H(x)$ is defined to be the number of nonzero components in x, as we defined earlier. If $R = \mathbb{Z}/m\mathbb{Z}$, the *Euclidean weight* is

$$\mathrm{wt}_E(x) = \sum_{i=1}^{n} \min\{x_i^2, (m - x_i)^2\},$$

and the *Lee weight* is

$$\mathrm{wt}_L(x) = \sum_{i=1}^{n} \min\{|x_i|, |m - x_i|\}.$$

As usual, the *distance* between two vectors x and y is $d_H(x, y) = \mathrm{wt}_H(x - y)$, $d_E(x, y) = \mathrm{wt}_E(x - y)$, or $d_L(x, y) = \mathrm{wt}_L(x - y)$, depending on if the weight function used is the Hamming, Euclidean, or Lee weight, respectively. The inner product of two vectors x and y is $\langle x, y\rangle = \sum_{i=1}^{n} x_i y_i$. The *dual code* $C^\perp$ of C is $\{y \in R^n | \langle x, y\rangle = 0 \text{ for all } \in C\}$. If $C \subset C^\perp$, C is called *self-orthogonal*, and if $C = C^\perp$, C is called *self-dual*. A self-dual code C over $\mathbb{Z}/2k\mathbb{Z}$ is called *even* (or *Type II*) if $\mathrm{wt}_E(x)$ is divisible by $4k$ for all $x \in C$ and *odd* otherwise.

Proposition 51 *Let $G(C)$ be a generator matrix of C over $\mathbb{Z}/2k\mathbb{Z}$. Suppose that all the rows x of $G(C)$ have $\mathrm{wt}_E(x)$ divisible by $4k$ and that any two rows of $G(C)$ are orthogonal. Then C is self-orthogonal with Euclidean weights that are multiples of $4k$.*

Proof By induction, it suffices to show that $4k$ divides $\mathrm{wt}_E(x + y)$ for any two rows x, y of $G(C)$. It is easy to see that $\mathrm{wt}_E(x + y) = \mathrm{wt}_E(x) + \mathrm{wt}_E(y) + 2(x_1 y_1 + \cdots + x_n y_n)$. Since $\langle x, y\rangle = \sum_{i=1}^{n} x_i y_i \pmod{2k}$, we have $4k | \mathrm{wt}_E(x + y)$. □

2.5 Lattices from Codes

Codes can be used to generate interesting lattices. In this section, we describe how to obtain lattices from linear codes over rings.

A (real) *lattice* is a free $\mathbb{Z}$-module Λ in $\mathbb{R}^n$ of full rank. For example, $\Lambda = Z^n \subset \mathbb{R}^n$. Let

$$\vec{g_1} = (g_{11}, \ldots, g_{1n}), \quad \ldots, \quad \vec{g_n} = (g_{n1}, \ldots, g_{nn}),$$

denote a $\mathbb{Z}$-basis of Λ. The n by n matrix whose rows are $\vec{g_1}, \ldots, \vec{g_n}$ is called the *generator matrix* M of Λ.

The *fundamental volume* $V(\Lambda)$ is the volume of the fundamental domain $\mathbb{R}^n/\Lambda$. It is well known that the absolute value of the determinant of the generator matrix and the fundamental volume are equal.

The *minimum squared distance* of Λ is

$$d(\Lambda) = \min_{x \in \Lambda, x \neq 0} x \cdot x = \min_{x \in \Lambda, x \neq 0} [x, x],$$

where

$$[x, y] = x \cdot y = \sum_{i=1}^{n} x_i y_i$$

is the usual dot product on $\mathbb{R}^n$. (We use the bracket notation in cases where ambiguity may arise.) For lattices, the "distance squared metric" is more convenient in some situations than the usual Euclidean metric. Let $A = MM^T$, called the *Gram matrix* of Λ.

Define $\det \Lambda := \det A = (\det M)^2$.

Let $O(n)$ denote the real orthogonal group of $n \times n$ matrices. A lattice Λ' is *equivalent* to Λ if there is an $A \in O(n)$ such that $\Lambda' = A\Lambda$. If $\Lambda = A\Lambda$, then A is called an *automorphism* of Λ. The subgroup of automorphisms is denoted $\mathrm{Aut}(\Lambda) \subset O(n)$.

The *dual lattice* is defined by

$$\Lambda^* = \left\{ y \in \mathbb{R}^n \mid x \cdot y \in \mathbb{Z} \text{ for all } x \in \Lambda \right\},$$

where $\cdot$ denotes the usual Euclidean inner product. (In the literature, Λ^* is also denoted $\Lambda^\perp$.) It is known that

$$\det \Lambda^\perp = (\det \Lambda)^{-1}.$$

A lattice for which $\Lambda = \Lambda^*$ is called *unimodular*. This is the lattice analog for a self-dual code. The *norm* $N(x)$ of $x \in \Lambda$ is defined by

$$N(x) = x \cdot x.$$

The *norm of* Λ is the subgroup

$$\mathcal{N}(\Lambda) = \left\langle N(x) \mid x \in \Lambda \right\rangle \subset \mathbb{R}.$$

If Λ is unimodular, then either $\mathcal{N}(\Lambda) = \mathbb{Z}$ or $\mathcal{N}(\Lambda) = 2\mathbb{Z}$.

A lattice Λ is called *integral* if

$$\Lambda \subset \Lambda^*.$$

In other words, Λ is integral if the inner product of any two vectors in the lattice is an integer. If $N(x)$ is an even integer for all $x \in \Lambda$, Λ is *even* (or *Type II*). A unimodular lattice Λ which is not even is called *Type I* or *odd*.

Proposition 52 *An integral lattice Λ satisfies $\Lambda \subset \Lambda^* \subset \frac{1}{\det \Lambda}\Lambda$.*

Proof Since Λ is integral, it suffices to show that $\Lambda^* \subset \frac{1}{\det \Lambda}\Lambda$. Let $x \in \Lambda^*$. Then $xM^T = \xi \in \mathbb{Z}^n$. Therefore, $x = \xi(M^T)^{-1} = \xi(M^T)^{-1}M^{-1}M = \xi A^{-1}M$. By Cramer's rule, $A^{-1} = \frac{1}{\det A}\operatorname{adj} A$. Hence,

$$\xi A^{-1}M = \xi\left(\frac{1}{\det A}\operatorname{adj} A\right)M = \frac{1}{\det A}(\xi \cdot \operatorname{adj} A)M = \frac{1}{\det A}\xi' M$$

for some $\xi' \in \mathbb{Z}$ as $\operatorname{adj} \in \mathbb{Z}$ has integer entries. Hence, $x \in \frac{1}{\det \Lambda}\Lambda$. □

Proposition 53 *An integral lattice Λ is unimodular if and only if* $\det \Lambda = 1$.

Proof If $\det \Lambda = 1$, then it follows from Proposition 52 that Λ is unimodular. Conversely, suppose that $\Lambda = \Lambda^*$. Since $\det \Lambda^* = \frac{1}{\det \Lambda}$ or equivalently $V(\Lambda^*) = \frac{1}{V(\Lambda)}$, $\det \Lambda = 1$, as required. □

How many inequivalent Type I lattices are there? The following remarkable "mass formula for lattices" gives an indication.

Theorem 54 *Let Ω be the set of inequivalent unimodular lattices of rank n which are Type I but not Type II* (*i.e., "odd"*). *Then*

$$\sum_{L \in \Omega} \left|\operatorname{Aut}(L)\right|^{-1}$$

equals

$$\frac{1}{2 \cdot (n/2)!} B_{n/2} B_2 B_4 \ldots B_{n-2}(1 - 2^{-n/2})(1 + 2^{(n-2)/2})$$

if $n \equiv 0 \pmod 8$. *Here B_k is the absolute value of the kth Bernoulli number.*

There are similar formulas for other values of n and for lattices of Type II. See Sect. 9 of Sloane's survey for details and references [Sl].

How "spread out" can a Type I or Type II lattice be? The following result is the analog of the Mallows–Sloane bound for self-dual binary codes.

Theorem 55 *If L is Type I, then*

$$d(\Lambda) \leq \left[\frac{n}{8}\right] + 1.$$

If Λ is Type II, then

$$d(\Lambda) \leq 2\left[\frac{n}{24}\right] + 2.$$

If the equality holds, then Λ is called an *extremal* lattice of that type.

2.5.1 Constructions from Codes

Constructing unimodular lattices has been one of the most interesting problems in Number Theory. Coding theory has played an important role in constructing unimodular lattices. In what follows we describe *Construction A*. This construction was introduced by Conway and Sloane over $\mathbb{Z}/2\mathbb{Z}$ [CS1], Bonnecaze, Solé, and Calderbank over $\mathbb{Z}/4\mathbb{Z}$ [BSC], and Bannai, Dougherty, Harada, and Oura over $\mathbb{Z}/2k\mathbb{Z}$ [BDHO].

Theorem 56 (Construction A) *If C is a self-dual code of length n over $\mathbb{Z}/2k\mathbb{Z}$, then the lattice*

$$\Lambda(C) = \frac{1}{\sqrt{2k}}\left\{x = (x_1, \ldots, x_n) \in \mathbb{Z}^n \mid \left(x_1 \pmod{2k}, \ldots, x_n \pmod{2k}\right) \in C\right\}$$

is an n-dimensional unimodular lattice with the minimum norm $\min\{\frac{d_E(C)}{2k}, 2k\}$. *Moreover, if C is Type II, then $\Lambda(C)$ is Type II.*

Proof Clearly $\Lambda(C)$ is an n-dimensional lattice. Let $a_1, a_2 \in \Lambda(C)$. Then $a_i = \frac{1}{\sqrt{2k}}(c_i + 2kz_i)$, where $c_i \in C$ and $z_i \in \mathbb{Z}^n$ for $i = 1, 2$. Then $[a_1, a_2] = \frac{1}{2k}([c_1, c_2] + 2k[c_1, z_2] + 2k[c_2, z_1] + 4k^2[z_1, z_2]) \in \mathbb{Z}^n$ since $[c_1, c_2]$ is a multiple of $2k$. Thus, $\Lambda(C)$ is integral. Next we show that $\Lambda(C)$ is unimodular. Note that $2k\mathbb{Z}^n \subset \sqrt{2k}\Lambda(C) \subset \mathbb{Z}^n$. It is easy to see that $V(2k\mathbb{Z}^n) = (2k)^n$ and $[\sqrt{2k}\Lambda(C) : 2k\mathbb{Z}^n] = (2k)^{n/2}$. Hence, $V(\sqrt{2k}\Lambda(C)) = (2k)^{n/2}$. Then $V(\Lambda(C)) = 1 = \det \Lambda$, that is, $\Lambda(C)$ is unimodular. Furthermore, $[a_1, a_1] \geq [c_1/\sqrt{2k}, c_1/\sqrt{2k}]$ for any $a_1 = (c_1 + 2kz_1)/\sqrt{2k}$. Hence, if $c_1 \neq 0$, then $[a_1, a_1] \geq d_E/2k$, and if $c_1 = 0$, then $[a_1, a_1] \geq 2k$. Therefore the minimum norm of $\Lambda(C)$ is $\min\{2k, d_E/2k\}$.

Suppose that C is type II. Then $[a_1, a_1] = \frac{1}{2k}([c_1, c_1] + 4k[c_1, z_1] + 4k^2[z_1, z_1]) \in 2\mathbb{Z}$ since $[c_1, c_1]$ is a multiple of $4k$. Therefore, $\Lambda(C)$ is Type II. □

Corollary 57 *Suppose that C is a self-dual code over $\mathbb{Z}/2k\mathbb{Z}$ such that every Euclidean weight is a multiple of a positive integer c. Then the largest positive integer c is $2k$ or $4k$.*

Proof By Construction A, $\Lambda(C)$ is unimodular. We use the fact that if a unimodular lattice has the property that every norm is a multiple of some positive integer d, then $d = 1$ or 2 (see O'Meara [OM]). Therefore, $c = 2k$ or $4k$. □

Generalizing Corollary 57, we suggest the following problem.

Open Problem 7 Suppose that C is a self-dual code over $\mathbb{Z}/m\mathbb{Z}$ such that every Euclidean weight is a multiple of a positive integer $b \geq 2$. Then one of the following holds.

(i) m is even, and b is either m or $2m$;
(ii) m is odd, and b is m.

Corollary 58 *There exists a Type II code C of length n over $\mathbb{Z}/2k\mathbb{Z}$ if and only if* 8 *divides n.*

Proof Suppose that there is a Type II code C. Then we have a Type II lattice $\Lambda(C)$. Then we use the fact that an n-dimensional unimodular lattice exists if and only if $8|n$.

For the converse, let I_4 be the 4×4 identity matrix, and let M_4 consist of four rows $(a,\ b,\ c,\ d)$, $(b,\ -a,\ -d,\ c)$, $(c,\ d,\ -a,\ -b)$, and $(d,\ -c,\ b,\ -a)$, where $a^2 + b^2 + c^2 + d^2 + 1 = 4k$, whose existence is guaranteed by Lagrange's sum of four squares. Then (I_4, M_4) generates a Type II code of length 8 over $\mathbb{Z}/2k\mathbb{Z}$. By constructing the direct sum of (I_4, M_4) we see that a Type II code of length a multiple of 8 exists. □

2.5.2 Theta Function of a Lattice

The *theta function* of an integral lattice Λ is

$$\theta_\Lambda(z) = \sum_{v \in \Lambda} q^{v \cdot v} = \sum_{r=0}^{\infty} a_r q^r,$$

where $q = e^{\pi i z}$ ($z \in \mathbb{C}$ and $\mathrm{Im}(z) > 0$), and

$$a_r = \big|\{v \in \Lambda \mid v \cdot v = r\}\big|.$$

The following result is the lattice analog of the MacWilliams identity:

$$\theta_{\Lambda^*}(z) = \sqrt{\det \Lambda} \cdot (i/z)^{n/2} \cdot \theta_\Lambda(-1/z). \tag{2.5.1}$$

The *zeta function* of the lattice is (more-or-less) obtained from the Mellin transform of $\theta_\Lambda(z) - 1$ along the imaginary axis:

$$\pi^{-s} \Gamma(s)(v \cdot v)^{-s} = \int_0^\infty x^{s-1} e^{-\pi (v \cdot v) x}\, dx,$$

so

$$\pi^{-s} \Gamma(s) \zeta_\Lambda(s) = \int_0^\infty x^{s-1} \big(\theta_\Lambda(ix) - 1\big)\, dx, \tag{2.5.2}$$

where $\zeta_\Lambda(s) = \sum_{v \in \Lambda, v \neq 0} (v \cdot v)^{-s}$. Note that if $\Lambda = \mathbb{Z}$, then $\zeta_\Lambda(s)$ is twice the Riemann zeta function, $2\zeta(s)$.

Remark 4 This Mellin transform computation of the zeta function of a lattice is analogous to the definition of the zeta function of a code.

Why? Recall that the Mellin transform is the continuous analog of the power series:

$$\int_0^\infty x^{s-1}a(x)\,dx \leftrightarrow \sum_{n=0}^{\infty} a(n)x^n.$$

Conversely, the inverse Mellin transform is analogous to extracting a power series coefficient.

In Chap. 2, we will define a polynomial $P(T)$ for which

$$\begin{aligned}\frac{(xT+(1-T)y)^n}{(1-T)(1-qT)}P(T) &= \cdots + \frac{A_C(x,y)-x^n}{q-1}T^{n-d}+\cdots\\ &= \cdots + x^n\frac{A_C(1,y/x)-1}{q-1}T^{n-d}+\cdots\\ &= \cdots + x^n\frac{W(z)-1}{q-1}T^{n-d}+\cdots,\end{aligned}$$

where $z = y/x$ and $W(z) = A_C(1, z)$. As we will see later, this polynomial $P(T)$ is the "Duursma zeta polynomial" of C, the numerator of a "Duursma zeta function." The theta-function of a lattice is analogous to the weight enumerator of a code. Since the Mellin transform is analogous to a power series, and the inverse Mellin transform is analogous to extracting a power series coefficient, we see that the zeta function associated to a code is analogous to the zeta function associated to a lattice. Indeed, in each case, the zeta function is a multiplicative factor of a "transform" of either $\theta - 1$ or $W - 1$.

Table of analogies

Binary codes	Lattices
C	$\Lambda = \Lambda(C)$
$A_C(x, y)$	$\theta_\Lambda(z)$
minimum distance, d	minimum squared distance $d(\Lambda)$
$\mathrm{Aut}(C) \subset S_n$	$\mathrm{Aut}(\Lambda) \subset O(n)$
$W_C(z)$	$\theta_\Lambda(z)$
$\zeta_C(T)$	$\zeta_\Lambda(s)$

Both zeta functions have a "Riemann hypothesis." See Chap. 2 for more details.

2.6 More Problems Related to a Prize Problem

In this section, we further describe one of the long-standing open problems in algebraic coding theory. This is about the existence of a binary self-dual [72, 36, 16] code. We refer to [Ki1].

Let C be a binary Type I code, and C_0 the doubly even subcode C_0 of C (that is, the subcode of C consisting of all codewords of weight $\equiv 0 \pmod 4$). Conway and

Sloane defined the *shadow* S of C by $S := C_0^{\perp} \backslash C$ [CS3]. The weight enumerator $A_S(x, y)$ of the shadow of C is determined by the weight enumerator $A_C(x, y)$ of C; $A_S(x, y) = \frac{1}{|C|} A_C(x + y, i(x - y))$, where $i = \sqrt{-1}$. This restricts a possible $W_C(x, y)$ of a binary self-dual code. In [CS3], Conway and Sloane gave possible weight enumerators of binary self-dual codes of lengths less than 72, and their work initiated a number of interesting research problems.

Huffman [Hu] gives the most updated status of the classification and enumeration of self-dual codes over $GF(2), GF(3), GF(4), \mathbb{Z}_4, GF(2) + uGF(2)$, and $GF(2) + vGF(2)$. Using shadow codes, Rains [Ra] derived a tight upper bound on the minimum distance of a binary Type I self-dual code. The minimum distance d of any binary self-dual code of length n satisfies $d \leq 4\lfloor n/24 \rfloor + 4$ except when $n \equiv 22 \pmod{24}$, in which case $d \leq 4\lfloor n/24 \rfloor + 6$ (see [Ra]). Further if C is a Type I code of length $n \equiv 0 \pmod{24}$, then $d \leq 4\lfloor n/24 \rfloor + 2$. Huffman calls a Type I self-dual code length $n \equiv 0 \pmod{24}$ whose minimum distance d attains this bound "extremal." This is a modification of the definition of Type I extremal codes given after Lemma 40 above. However, for the purposes of stating some of the results in Chap. 4 in the "cleanest way," we retain the "unmodified" definition of extremal in this work. In other words, we say that a self-dual code C is *extremal* if equality is attained in the "original" Mallows–Sloane bounds.

Open Problem 8 Prove or disprove that there exists a Type II $[24k, 12k, 4k + 4]$-code $C(k)$ for $k \geq 3$.

We note that $C(1)$ is the binary extended Golay code; any binary linear code with parameters $[24, 12, 8]$ is equivalent to $C(1)$ [Pl]. Further, $C(2)$ is the extended quadratic residue code XQ_{47} of length 48. It was recently shown that this is unique up to equivalence among self-dual codes with parameters $[48, 24, 12]$ [HLTP]. This raises the following question.

Open Problem 9 Prove or disprove that there is a binary linear $[48, 24, 12]$ code other than XQ_{47}.

The existence of $C(3)$ is the first unknown case in the family $C(k)$. It was given as Open Problem 6. This was originally suggested by Sloane in 1973 [S73]. The existence of $C(3)$ will imply a 5-$(72, 16, 78)$ design, whose existence is still unknown. By Theorem 41, $C(k)$ does not exist for $k \geq 154$ since A_{4k+8} is negative [Z].

The weight enumerator of a putative Type II $[72, 36, 16]$ code $C(3)$ is given:

$$\begin{aligned} W_{C(3)}(1, y) = 1 &+ 249,849y^{16} + 18,106,704y^{20} + 462,962,955y^{24} \\ &+ 4,397,342,400y^{28} + 16,602,715,899y^{32} \\ &+ 25,756,721,120y^{36} + \cdots . \end{aligned}$$

It is known that the only possible *prime orders* of an automorphism of $C(3)$ are 2, 3, 5, and 7. Yorgov [Hu] proved that the automorphism group has order a divisor of 72 or order 504, 252, 56, 14, 7, 360, 180, 60, 30, 10, or 5.

The existence of $C(3)$ is equivalent to that of a Type I $[70, 35, 14]$ code [Ra]. The weight enumerator of a Type I $[70, 35, 14]$ code is corrected in [Hu]:

$$W = 1 + 11,730y^{14} + 150,535y^{16} + 1,345,960y^{18} + \cdots .$$

It is shown [GHK] that the existence of $C(k)$ implies the existence of a Type I $[24k, 12k, 4k + 2]$ code for $k \geq 1$. Hence the existence of $C(3)$ implies a Type I $[72, 36, 14]$ code. Equivalently, the nonexistence of a Type I $[72, 36, 14]$ code implies the nonexistence of $C(3)$. There is no known self-dual code with parameters $[72, 36, 14]$. There are exactly three possible weight enumerators for a Type I $[72, 36, 14]$ code:

$$W_1 = 1 + 7616y^{14} + 134,521y^{16} + 1,151,040y^{18} + \cdots ,$$
$$W_2 = 1 + 8576y^{14} + 124,665y^{16} + 1,206,912y^{18} + \cdots ,$$
$$W_3 = 1 + 8640y^{14} + 124,281y^{16} + 1,207,360y^{18} + \cdots .$$

Open Problem 10 Prove or disprove that there is a binary linear $[72, 36, 16]$ code.

We note that there is a $[72, 36, 15]$ code by puncturing a $[73, 36, 16]$ cyclic code and that any $[72, 36, d]$ code satisfies $d \leq 17$ from Brouwer's Table.

As far as we know, the existence of $C(3)$ is the only coding problem with monetary prizes.

- N.J.A. Sloane offers \$10 (1973)—still valid.
- F.J. MacWilliams offered \$10 (1977)—invalid now.
- S.T. Dougherty offers \$100 for the existence of $C(3)$.
- M. Harada offers \$200 for the nonexistence of $C(3)$.

Further information of $C(3)$ can be found in Dougherty's website:

http://academic.scranton.edu/faculty/doughertys1/.

Chapter 3
Kittens, Mathematical Blackjack, and Combinatorial Codes

Can coding theory help you be a better gambler? Do unexpected combinatorial patterns (called Steiner systems) arise naturally in codes? This chapter attempts to address these types of questions.[1]

This chapter gives an exposition of some ideas of Hadamard and Mathieu, as well as ideas of Conway, Curtis, and Ryba connecting the Steiner system $S(5, 6, 12)$ with a card game called mathematical blackjack. An implementation in SAGE is described as well. Then we turn to one of the most beautiful results in coding theory, the Assmus–Mattson Theorem, which relates certain linear codes to combinatorial structures called designs. Finally, we describe an old scheme for placing bets using Golay codes (the scheme was in fact published in a Finnish soccer magazine years before the error-correcting code itself was discovered [HH]).

Open questions which arise in this chapter include conjectures on Hadamard matrices and on which block designs "arise" from an error-correcting code.

3.1 Hadamard Matrices and Codes

Let $A = (a_{ij})_{1 \le i,j \le n}$ be a real $n \times n$ matrix. The following question seems quite natural in a course in advanced vector calculus or real analysis. It is not solved in general and is called the Hadamard maximum determinant problem.[2]

[1] The books by Assmus and Key [AK] and Conway and Sloane [CS1] do into more detail in these matters and are highly recommended.

[2] Hadamard determined the maximum value of $|\det(A)|$, where the entries of A range over all *complex* numbers $|a_{ij}| \le 1$, to be $n^{n/2}$ and that this maximum was attained by the Vandermonde matrices of the nth roots of unity.

D. Joyner, J.-L. Kim, *Selected Unsolved Problems in Coding Theory*,
Applied and Numerical Harmonic Analysis,
DOI 10.1007/978-0-8176-8256-9_3,

Open Problem 11 What is the maximum value of $|\det(A)|$, where the entries of A range over all real numbers $|a_{ij}| \leq 1$?

From vector calculus we know that the absolute value of the determinant of a real square matrix equals the volume of the parallelepiped spanned by the row (or column) vectors of the matrix. The volume of a parallelepiped with sides of a *fixed* length depends on the angles the row vectors make with each other. This volume is maximized when the row vectors are mutually orthogonal, i.e., when the parallelepiped is a cube in $\mathbb{R}^n$. Suppose now that the row vectors of A are all orthogonal. The row vectors of A, $|a_{ij}| \leq 1$, are longest when each $a_{ij} = \pm 1$, which implies that the length of each row vector is $\sqrt{n}$. Suppose, in addition, that the row vectors of A are all of length $\sqrt{n}$. Such a matrix is called a *Hadamard matrix of order n*. Then $|\det(A)| = (\sqrt{n})^n = n^{n/2}$, since the cube has n sides of length $\sqrt{n}$. Now, if A is any matrix as in the above question, then we must have $|\det(A)| \leq n^{n/2}$. This inequality is called *Hadamard's determinant inequality*.

Jacques Hadamard (1865–1963) was a prolific mathematician who worked in many areas, but he is most famous for giving one of the first proofs of the prime number theorem (in 1896). (The prime number theorem, "known" to Gauss though not proven, roughly states that the number of prime numbers less than N is about $N/\log(N)$ as N grows to infinity.)

The above question is unsolved for arbitrary n in the sense that it is not yet (as of this writing) known for which n Hadamard matrices exist. Moreover, if $n \equiv 3 \pmod 4$ (for such n, it is known that Hadamard matrices cannot exist), then the best possible upper bound for $|\det(A)|$, as A ranges over all $n \times n$ $(-1, 1)$-matrices, is not known at the time of this writing.

Open Problem 12 (Hadamard conjecture) For each n which is a multiple of 4, a Hadamard matrix exists.

(This conjecture might, in fact, be due to Raymond Paley, a brilliant mathematician who contributed to several areas of mathematics, including two constructions of Hadamard matrices using the theory of finite fields. Sadly, he died in 1933 at the age of 26 in a skiing accident.) At the time of this writing, the smallest order for which no Hadamard matrix is presently known is 668.

What might be surprising at first sight is that there does not always exist a Hadamard matrix—for some n, they exist, and for other n, they do not. For example, there is a 2×2 Hadamard matrix but not a 3×3 one. What is perhaps even more surprising is that, in spite of the fact that the above question arose from an analytic perspective, Hadamard matrices are related more to coding theory, number theory, and combinatorics [vLW], [Ho]! In fact, a linear code constructed from a Hadamard matrix was used in the 1971 Mariner 9 mission to Mars.

Example 59 Let

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 \\ -1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 \\ -1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 \\ -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 \\ -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 \\ -1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 \\ -1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 \\ -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 \end{pmatrix}.$$

This is a Hadamard matrix of order 12. A different Hadamard matrix of order 12 can be computed using SAGE:

SAGE

```
sage: hadamard_matrix(12)

[ 1  1  1  1  1  1  1  1  1  1  1  1]
[ 1 -1  1 -1  1  1  1 -1 -1 -1  1 -1]
[ 1 -1 -1  1 -1  1  1  1 -1 -1 -1  1]
[ 1  1 -1 -1  1 -1  1  1  1 -1 -1 -1]
[ 1 -1  1 -1 -1  1 -1  1  1  1 -1 -1]
[ 1 -1 -1  1 -1 -1  1 -1  1  1  1 -1]
[ 1 -1 -1 -1  1 -1 -1  1 -1  1  1  1]
[ 1  1 -1 -1 -1  1 -1 -1  1 -1  1  1]
[ 1  1  1 -1 -1 -1  1 -1 -1  1 -1  1]
[ 1  1  1  1 -1 -1 -1  1 -1 -1  1 -1]
[ 1 -1  1  1  1 -1 -1 -1  1 -1 -1  1]
[ 1  1 -1  1  1  1 -1 -1 -1  1 -1 -1]
```

Here SAGE is calling a native Python program, which implements some methods for constructing Hadamard matrices. Another way to use SAGE to obtain a Hadamard matrix is to look up the file name on Sloane's database http://www.research.att.com/~njas/hadamard/, for example, `had.16.2.txt`, and use the SAGE command

SAGE

```
sage: hadamard_matrix_www('had.16.2.txt')

[ 1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1]
[ 1 -1  1 -1  1 -1  1 -1  1 -1  1 -1  1 -1  1 -1]
```

```
[ 1  1 -1 -1  1  1 -1 -1  1  1 -1 -1  1  1 -1 -1]
[ 1 -1 -1  1  1 -1 -1  1  1 -1 -1  1  1 -1 -1  1]
[ 1  1  1  1 -1 -1 -1 -1  1  1  1  1 -1 -1 -1 -1]
[ 1 -1  1 -1 -1  1 -1  1  1 -1  1 -1 -1  1 -1  1]
[ 1  1 -1 -1 -1 -1  1  1  1  1 -1 -1 -1 -1  1  1]
[ 1 -1 -1  1 -1  1  1 -1  1 -1 -1  1 -1  1  1 -1]
[ 1  1  1  1  1  1  1  1 -1 -1 -1 -1 -1 -1 -1 -1]
[ 1  1  1  1 -1 -1 -1 -1 -1 -1 -1 -1  1  1  1  1]
[ 1  1 -1 -1  1 -1  1 -1 -1 -1  1  1 -1  1 -1  1]
[ 1  1 -1 -1 -1  1 -1  1 -1 -1  1  1  1 -1  1 -1]
[ 1 -1  1 -1  1 -1 -1  1 -1  1 -1  1 -1  1  1 -1]
[ 1 -1  1 -1 -1  1  1 -1 -1  1 -1  1  1 -1 -1  1]
[ 1 -1 -1  1  1  1 -1 -1 -1  1  1 -1 -1 -1  1  1]
[ 1 -1 -1  1 -1 -1  1  1 -1  1  1 -1  1  1 -1 -1]
```

This command assumes that you are running SAGE from a computer with an internet connection since it actually fetches the file `had.16.2.txt` from http://www research.att.com/~njas/hadamard/ and parses that file to return the matrix given above.

Some easy to prove facts:

(a) if you swap two rows or columns of a Hadamard matrix, you will get another Hadamard matrix;
(b) if you multiply any row or column of a Hadamard matrix by -1, you will get another Hadamard matrix;
(c) if you multiply any Hadamard matrix on the left by a signed permutation matrix (that is, a matrix with exactly one ± 1 per row and column), you will get another Hadamard matrix;
(d) if you multiply any Hadamard matrix on the left by a signed permutation matrix (that is, a matrix with exactly one ± 1 per row and column), you will get another Hadamard matrix.

Definition 60 Let A, B be two Hadamard matrices of order n. Call A and B *left equivalent* if there is an $n \times n$ signed permutation matrix P such that $A = PB$. Let A be a Hadamard matrix of order n. Let $\mathrm{Aut}(A)$ denote the group of all $n \times n$ signed permutation matrices Q such that A is left equivalent to AQ. This object $\mathrm{Aut}(A)$ is called the *automorphism group of* A.

Mathieu groups, discovered in the 1800's, are now know to arise natuarally in many fields of mathematics (see Conway-Sloane [CS1] for an excellent treatment). The following result is just one indication of the unique role of Mathieu groups in mathematics.

Theorem 61 *It is known that any two* 12×12 *Hadamard matrices are equivalent, i.e., that there is only one Hadamard matrix of order* 12*, up to equivalence. Let* A *be a* 12×12 *Hadamard matrix. Then* $\mathrm{Aut}(A) \cong M_{12}$.

The proof can be found in Assmus and Mattson [AM1] or in Sect. 7.4 in Assmus and Key [AK] (see in particular their Theorem 7.4.3, which also discusses more general codes associated to Hadamard matrices). Kantor [Kan] is another excellent paper on this topic.

3.2 Designs, Orthogonal Arrays, Latin Squares, and Codes

An *m-(sub)set* is a (sub)set with m elements. For integers $k < m < n$, a *Steiner system* $S(k, m, n)$ is an n-set X and a set S of m-subsets having the property that any k-subset of X is contained in exactly one m-set in S. For example, if $X = \{1, 2, \ldots, 12\}$, a Steiner system $S(5, 6, 12)$ is a set of 6-sets, called *hexads*, with the property that any set of 5 elements of X is contained in ("can be completed to") exactly one hexad.

A t-(v, k, λ) *design* $D = (P, B)$ is a pair consisting of a set P of *points* and a collection B of k-element subsets of P, called *blocks*, such that the number r of blocks that contain any point $p \in P$ is independent of p, and the number λ of blocks that contain any given t-element subset T of P is independent of the choice of T. The numbers v (the number of elements of P), b (the number of blocks), k, r, λ, and t are the parameters of the design. The parameters must satisfy several combinatorial identities, such as

$$\lambda_i = \lambda \binom{v-i}{t-i} \Big/ \binom{k-i}{t-i}$$

where λ_i is the number of blocks that contain any i-element set of points ($1 \le i \le t$). A design without repeated blocks is called a *simple* design. A design with $t = 2$ is called a *(balanced incomplete) block design* (ir BIBD).

We may think of elements of $GF(q)^n$ as both points and blocks as follows: let $b = (b_1, \ldots, b_n) \in GF(q)^n$ be regarded as a block, and $p = (p_1, \ldots, p_n) \in GF(q)^n$ be regarded as a point. We say that *b covers p*, or *p* is *in b*, provided that (a) $\mathrm{supp}(p) \subset \mathrm{supp}(b)$ and (b) for all $i \in \mathrm{supp}(b)$, (at least) one of the following conditions holds: (i) $p_i = b_i$, (ii) $p_i = 0$.

More generally, a q-*ary* t-(v, k, λ) *design* $D = (P, B)$ is a pair consisting of the set $P \subset GF(q)^n$ of elements (called *points*) of weight t and a collection B of weight k elements of $GF(q)^n$ (called *blocks*) such that every point $p \in P$ is covered by exactly λ blocks.

Example 62 Consider the 3-ary Hamming code having the generator matrix

$$G = \begin{pmatrix} 1 & 0 & 2 & 2 \\ 0 & 1 & 2 & 1 \end{pmatrix}.$$

The nonzero codewords are

$$B = \big\{(1,0,2,2),(2,0,1,1),(0,1,2,1),(1,1,1,0),(2,1,0,2),$$
$$(0,2,1,2),(1,2,0,1),(2,2,2,0)\big\}.$$

Let P denote the elements of $GF(3)^4$ having weight 2. There are 24 elements in P. Each of these vectors is covered by exactly one element of B. Therefore, B, P gives rise to a 3-ary 2-$(4,3,1)$ design.

A *Steiner system* $S(t,k,v)$ is a t-(v,k,λ) design with $\lambda = 1$. A *Steiner triple system* is a Steiner system of the form $S(2,3,v)$.

Example 63 Here is a SAGE example.

SAGE

```
sage: from sage.combinat.designs.block_design import steiner_triple_system
sage: sts = steiner_triple_system(9)
sage: sts
Incidence structure with 9 points and 12 blocks
sage: sts.incidence_matrix()
[1 1 1 1 0 0 0 0 0 0 0 0]
[1 0 0 0 1 1 1 0 0 0 0 0]
[0 1 0 0 1 0 0 1 1 0 0 0]
[0 0 1 0 1 0 0 0 0 1 1 0]
[0 1 0 0 0 1 0 0 0 1 0 1]
[1 0 0 0 0 0 0 1 0 0 1 1]
[0 0 1 0 0 0 1 0 1 0 0 1]
[0 0 0 1 0 1 0 0 1 0 1 0]
[0 0 0 1 0 0 1 1 0 1 0 0]
sage: sts.points()
[0, 1, 2, 3, 4, 5, 6, 7, 8]
```

Open Problem 13 (Assmus–Mattson [AM2]) What single-error-correcting linear codes have the property that the supports of the codewords of weight 3 form a Steiner triple system?

Remark 1 Example 62 above gives us one example. See Assmus–Mattson [AM2] and the excellent survey by van Lint [vL2] for more details on this open question.

There are no known Steiner systems $S(t,k,v)$ with $t > 5$. The only ones known (at the time of this writing) for $t = 5$ are as follows:

$S(5,6,12)$, $S(5,6,24)$, $S(5,8,24)$, $S(5,7,28)$, $S(5,6,48)$, $S(5,6,72)$, $S(5,6,84)$, $S(5,6,108)$, $S(5,6,132)$, $S(5,6,168)$, and $S(5,6,244)$.

Open Problem 14 Are there others with $t = 5$? Are there any with $t > 5$?

In Assmus–Mattson [AM2], there is mentioned a conjecture of Peyton Young on the existence of a Steiner system with $t = 9$. This is still open, as far as we know.

3.2.1 Examples from Golay Codes

This subsection focuses on Steiner systems supporting Golay codes, such as $S(5, 6, 12)$ and $S(5.8.24)$.

If S is a Steiner system of type $(5, 6, 12)$ in a 12-set X, then the symmetric group S_X of X sends S to another Steiner system $\sigma(S)$ of X. It is known that if S and S' are any two Steiner systems of type $(5, 6, 12)$ in X, then there is a $\sigma \in S_X$ such that $S' = \sigma(S)$. In other words, a Steiner system of this type is unique up to relabelings. (This also implies that if one defines M_{12} to be the stabilizer of a fixed Steiner system of type $(5, 6, 12)$ in $X = \{1, 2, \ldots, 12\}$, then any two such groups, for different Steiner systems in X, must be conjugate in S_X. In particular, such a definition is well defined up to isomorphism.)

By means of an example of how designs arise naturally from codes, we recall some facts (see Conway and Sloane [CS1] and Sect. 8.4, page 303 of Huffman and Pless [HP1]).

Recall that the extended binary Golay code is a linear code over $GF(2)$ which is of length 24, dimension 12, and minimum distance 8. Let C be an $[n, k, d]$ code, and let

$$C_i = \left\{ c \in C \mid wt(c) = i \right\}$$

denote the weight i subset of codewords of weight i.

Lemma 64 (Conway, Assmus, and Mattson) *Let C denote the extended binary Golay code. With respect to a fixed basis, coordinates supporting the codewords of weight* 8 *form the* 759 *octads of a Steiner system $S(5, 8, 24)$. More generally, The set $X_8 = \{\text{supp}(c) \mid c \in C_8\}$ is a* 5-$(24, 8, 1)$ *design* (*which is a Steiner system*); *$X_{12} = \{\text{supp}(c) \mid c \in C_{12}\}$ is a* 5-$(24, 12, 48)$ *design; and $X_{16} = \{\text{supp}(c) \mid c \in C_{16}\}$ is a* 5-$(24, 16, 78)$ *design.*

Recall that the extended ternary Golay code is a linear code over $GF(3)$ of length 12 which is dimension 6 and minimum distance 6.

Lemma 65 (Conway) *Let C denote the extended ternary Golay code. With respect to a fixed basis, coordinates supporting the codewords of weight* 6 *form the* 132 *hexads of a Steiner system $S(5, 8, 24)$. Conversely, for each hexad in this Steiner system, there are exactly two codewords in C_{12} supported on that hexad.*

3.2.2 Assmus–Mattson Theorem

The previous section looked at the designs arising from the supports of codewords of a fixed weight form a Golay code. More generally, we have the following theorem of Assmus and Mattson, which gives simple conditions under which a more general code can produce analogous designs. Not only is the theorem below one

of the most remarkable results in coding theory, it is helpful when computing the automorphism group of a such a code and therefore can be useful in permutation decoding procedures.

Theorem 66 (Assmus–Mattson Theorem[3]) *Let $A_0, A_1, \ldots, A_n$ be the weight distribution of the codewords in a binary linear $[n, k, d]$ code C, and let $A_0^\perp, A_1^\perp, \ldots, A_n^\perp$ be the weight distribution of the codewords in its dual $[n, n-k, d^\perp]$ code $C^\perp$. Fix a t, $0 < t < d$, and let $s = |\{i \mid A_i^\perp \neq 0, 0 < i \leq n - t\}|$. Assume that $s \leq d - t$.*

- *If $A_i \neq 0$ and $d \leq i \leq n$, then $C_i = \{c \in C \mid \mathrm{wt}(c) = i\}$ holds a simple t-design.*
- *If $A_i^\perp \neq 0$ and $d^\perp \leq i \leq n - t$, then $C_i^\perp = \{c \in C^\perp \mid \mathrm{wt}(c) = i\}$ holds a simple t-design.*

Remark 2

- There is an interesting strengthening of this result in the paper of Calderbank, Delsarte, and Sloane [CDS].
- The paper of Koch [Koch] (elucidating a theorem of Venkov) provides a fascinating addendum to this result in the special case where the code C is an extremal self-dual binary code.

In the Assmus and Mattson Theorem, X is the set $\{1, 2, \ldots, n\}$ of coordinate locations and $B = \{\mathrm{supp}(c) \mid c \in C_i\}$ is the set of supports of the codewords of C of weight i. Therefore, the parameters of the t-design for C_i are

- t (given),
- $v = n$,
- $k = i$ (this k is not to be confused with $\dim(C)$, of course!),
- $b = A_i$,
- $\lambda = b \cdot \dfrac{\binom{k}{t}}{\binom{v}{t}}$

(by Theorem 8.1.6, p. 294, in [HP1]).

Example 67 Consider the extended binary Hamming code with parameters $[8, 4, 4]$. This is a self-dual binary code. Take $k = 4$, $q = 2$, and $v = 4$, There are 14 codewords of weight 4. These 14 codewords form the blocks of a simple 3-$(8, 4, 1)$ design.

Example 68 Consider the extended binary quadratic residue code with parameters $[48, 24, 12]$. This is a self-dual binary code. The codewords of weight 12 form a 5-$(48, 12, 8)$ design.

[3] See the excellent survey article by van Lint [vL2] or Sect. 8.4, p. 303 of [HP1].

Here is a SAGE computation supporting this example.

SAGE

```
sage: C = ExtendedQuadraticResidueCode(47,GF(2))
sage: C
Linear code of length 48, dimension 24 over Finite Field of size 2
sage: C.is_self_dual()
True
sage: C.minimum_distance()
12
sage: C.assmus_mattson_designs(5)
['weights from C: ', [12, 16, 20, 24, 28, 32, 36, 48],
 'designs from C: ',
 [[5, (48, 12, 8)], [5, (48, 16, 1365)], [5, (48, 20, 36176)],
  [5, (48, 24, 190680)], [5, (48, 28, 229320)], [5, (48, 32, 62930)],
  [5, (48, 36, 3808)], [5, (48, 48, 1)]],
 'weights from C*: ', [12, 16, 20, 24, 28, 32, 36],
 'designs from C*: ',
 [[5, (48, 12, 8)], [5, (48, 16, 1365)], [5, (48, 20, 36176)],
  [5, (48, 24, 190680)], [5, (48, 28, 229320)], [5, (48, 32, 62930)],
  [5, (48, 36, 3808)]]]
```

This example is also discussed in Assmus–Mattson [AM2], Sect. III.

Example 69 Here is another SAGE example.

SAGE

```
sage: C = ExtendedBinaryGolayCode()
sage: C.assmus_mattson_designs(5)
['weights from C: ',
[8, 12, 16, 24],
'designs from C: ',
[[5, (24, 8, 1)], [5, (24, 12, 48)], [5, (24, 16, 78)], [5, (24, 24, 1)]],
'weights from C*: ',
[8, 12, 16],
'designs from C*: ',
[[5, (24, 8, 1)], [5, (24, 12, 48)], [5, (24, 16, 78)]]]
sage: C.assmus_mattson_designs(6)
0
sage: blocks = [c.support() for c in C if hamming_weight(c)==8]; len(blocks)
759
```

The automorphism group of the extended binary Golay code is the Mathieu group M_{24}. Moreover, the code is spanned by the codewords of weight 8.

Janusz have improved the Assmus–Mattson Theorem in the case of binary extremal Type II codes as follows [Ja2].

Theorem 70 [Ja2] *Let C be a $[24m+8\mu, 12m+4\mu, 4m+4]$ extremal Type II code for $\mu = 0, 1,$ or 2, where $m \geq 1$ if $\mu = 0$, and $\mu \geq 0$ otherwise. Then only one of the following holds*:

(a) *the codewords of any fixed weight $i \neq 0$ hold t-designs for $t = 7 - 2\mu$, or*
(b) *the codewords of any fixed weight $i \neq 0$ hold t-designs for $t = 5 - 2\mu$, and there is no i with $0 < i < 24m + 8\mu$ such that codewords of weight i hold a $(6 - 2\mu)$-design.*

As mentioned in [HP1], there are no known extremal Type II codes where the second part of Janusz's theorem is false. In particular, if $n = 24m$, this theorem implies that if the codewords of some fixed weight i with $0 < i < n$ hold a 6-design, then the codewords of any fixed weight $i \neq 0$ should hold 7-design. This indicates that it is hard to obtain t-designs in codes for $t > 5$. In fact, there is no known example of t-designs in *codes* if $t > 5$. However, there do exist t-designs for any $t > 5$ (Tierlinck [Tei]).

The following question is more specific than that in Open Question 14.

Open Problem 15 Does there exist a nontrivial 6-design held by the codewords of a fixed weight in a code?

3.2.3 Orthogonal Arrays, Latin Squares and Codes

An $M \times n$ matrix A with entries from a set of q elements is called an *orthogonal array of size M, having n constraints, q levels, strength k, and with index λ*, if any set of k columns of A contains all q^k possible row vectors exactly λ times. Such an array is denoted by $\mathrm{OA}(M, n, q, k)$. See [HSS] for details.

A *Latin square of order n* is an $n \times n$ array in which n distinct symbols are arranged so that each symbol occurs exactly once in each row and column.

Example 71 An example of a Latin square is

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 & 0 \\ 2 & 3 & 4 & 0 & 1 \\ 3 & 4 & 0 & 1 & 2 \\ 4 & 0 & 1 & 2 & 3 \end{pmatrix}.$$

Latin squares can be constructed using the multiplication table of a group. For instance, the dihedral group D_4 gives rise to

$$\begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 3 & 2 \\ 2 & 3 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{pmatrix},$$

and the symmetric group S_3 gives rise to

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 \\ 1 & 0 & 3 & 2 & 5 & 4 \\ 2 & 4 & 0 & 5 & 1 & 3 \\ 3 & 5 & 1 & 4 & 0 & 2 \\ 4 & 2 & 5 & 0 & 3 & 1 \\ 5 & 3 & 4 & 1 & 2 & 0 \end{pmatrix}.$$

Here are some examples using SAGE.

SAGE

```
sage: from sage.combinat.matrices.latin import *
sage: B = back_circulant(5); B
[0 1 2 3 4]
[1 2 3 4 0]
[2 3 4 0 1]
[3 4 0 1 2]
[4 0 1 2 3]
sage: B.is_latin_square()
True
sage: L = group_to_LatinSquare(DihedralGroup(2)); L
[0 1 2 3]
[1 0 3 2]
[2 3 0 1]
[3 2 1 0]
sage: L = group_to_LatinSquare(SymmetricGroup(3)); L
[0 1 2 3 4 5]
[1 0 3 2 5 4]
[2 4 0 5 1 3]
[3 5 1 4 0 2]
[4 2 5 0 3 1]
[5 3 4 1 2 0]
```

Let $L_1 = (a_{ij})$ and $L_2 = (b_{ij})$ be Latin squares of order $n \geq 2$. The arrays L_1 and L_2 are called *mutually orthogonal* if, when superimposed into the $n \times n$ array of pairs $((a_{ij}, b_{ij}))$, each of the possible n^2 ordered pairs occur exactly once. A set $\{L_1, \ldots, L_r\}$ of r Latin squares of order $n \geq 2$ is said to be *orthogonal* if any two distinct Latin squares are orthogonal. We call this a set of mutually orthogonal Latin squares (MOLS).

Example 72 An example of MOLS are

$$\begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 3 & 2 \\ 2 & 3 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{pmatrix}, \qquad \begin{pmatrix} 0 & 2 & 3 & 1 \\ 1 & 3 & 2 & 0 \\ 2 & 0 & 1 & 3 \\ 3 & 1 & 0 & 2 \end{pmatrix}.$$

Let $N(n)$ denote the maximum possible number of MOLS of order n. It is well known that $N(n) \leq n - 1$ for any $n \geq 2$ and $N(n) \geq 2$ for any n except 2 and 6. See the book [LM] on Latin squares for details.

Theorem 73 [MS] *The rows of an* OA(q^k, n, q, k) *linear orthogonal array A of index unity and symbols from $GF(q)$ are the codewords of an $[n, k]$ MDS code over $GF(q)$, and conversely.*

MacWilliams and Sloane [MS] suggest the following problem.

Open Problem 16 Find the greatest possible n in an $\mathrm{OA}(q^k, n, q, k)$ of index unity.

Orthogonal arrays and Latin squares are related as follows.

Theorem 74 *An MDS code with $k = 2$ is equivalent to a set of $n - k$ mutually orthogonal Latin squares of order q.*

This was proven by Golomb, Posner, and Singleton (see [Sin] for a proof).
For $1 < n \leq 99$, the number of MOLS of order n is given by the table

2	3	4	5	6	7	8	9
1	2	3	4	1	6	7	8

It is conjectured that $N(10) = 2$. It is known that $N(11) = 10$ and $N(12) \geq 5$. See Sequence A001438 in OEIS [OEIS]. In general, the exact value of $N(n)$ is unknown.

Open Problem 17 Determine $N(n)$ as the number of MOLS for $n \geq 10$.

3.3 Curtis' Kitten, Conway's Minimog

Conway and Curtis [Cu1] found a relatively simple and elegant way to construct hexads in a particular Steiner system $S(5, 6, 12)$ using the arithmetical geometry of the projective line over the finite field with 11 elements. This section describes this.

Let

$$\mathbf{P}^1\big(GF(11)\big) = \{\infty, 0, 1, 2, \ldots, 9, 10\}$$

denote the projective line over the finite field $GF(11)$ with 11 elements. Let

$$Q = \{0, 1, 3, 4, 5, 9\}$$

denote the quadratic residues and 0, and let

$$L = \langle \alpha, \beta \rangle \cong PSL\big(2, GF(11)\big),$$

where $\alpha(y) = y + 1$ and $\beta(y) = -1/y$. Let

$$S = \big\{\lambda(Q) \mid \lambda \in L\big\}.$$

Lemma 75 *S is a Steiner system of type* $(5, 6, 12)$.

The elements of S are known as *hexads* (in the "modulo 11 labeling").

The "views" of Curtis' kitten (Fig. 3.1) from each of the three "points at infinity" ($\{0, 1, \infty\}$ are the "points at infinity" in the "modulo 11 labeling) is given in the following tables.

6	10	3
2	7	4
5	9	8

picture at ∞

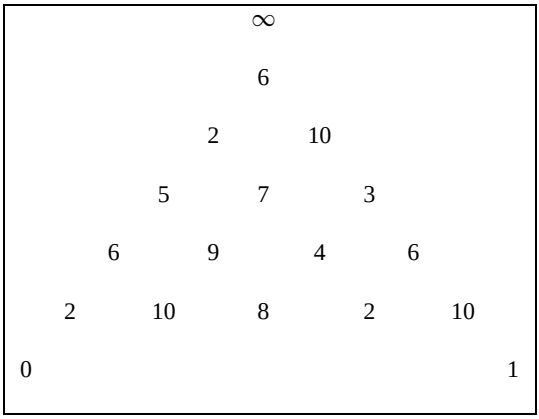
5	7	3
6	9	4
2	10	8

picture at 0

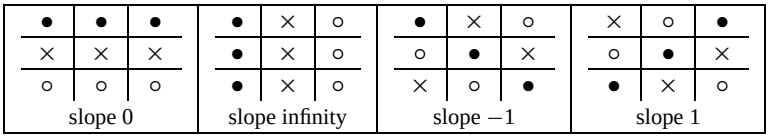
5	7	3
9	4	6
8	2	10

picture at 1

Fig. 3.1 Curtis' Kitten



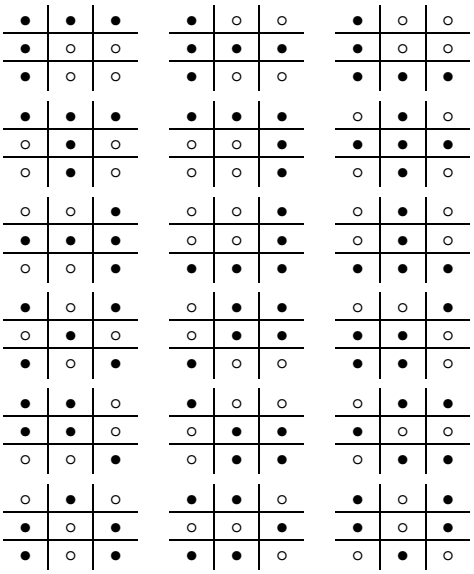
Each of these 3×3 arrays may be regarded as the plane $GF(3)^2$. The *lines* of this plane are described by one of the following patterns.



SAGE

```
sage: from sage.games.hexad import view_list
sage: M = Minimog(type="modulo11")
sage: view_list(M.line[0])

[1 1 1]
[0 0 0]
[0 0 0]
sage: view_list(M.line[3])

[0 0 1]
[0 0 1]
[0 0 1]
sage: view_list(M.line[6])

[1 0 0]
[0 1 0]
[0 0 1]
sage: view_list(M.line[9])

[0 0 1]
[0 1 0]
[1 0 0]
```

The union of any two perpendicular lines is called a *cross*. There are 18 crosses. The *crosses* of this plane are described by one of the following patterns of filled circles.



The complement of a cross in $GF(3)^2$ is called a *square*. Of course, there are also 18 squares, and the squares of this plane are described by one of the above patterns of hollow circles.

The hexads are:

1. three "points at infinity" union any line;
2. the union of any two (distinct) parallel lines in the same picture;
3. one "point at infinity" union a cross in the corresponding picture;
4. two "points at infinity" union a square in the picture corresponding to the omitted point at infinity.

SAGE

```
sage: from sage.games.hexad import view_list
sage: M = Minimog(type="modulo11")
sage: view_list(M.cross[0])

[1 1 1]
[1 0 0]
[1 0 0]
sage: view_list(M.cross[10])

[0 0 1]
[1 1 0]
[1 1 0]
```

Lemma 76 (Curtis [Cu1]) *There are* 132 *such hexads* (12 *of type* 1, 12 *of type* 2, 54 *of type* 3, *and* 54 *of type* 4). *They form a Steiner system of type* $(5, 6, 12)$.

3.3.1 The MINIMOG Description

Following Curtis' description [Cu2] of a Steiner system $S(5, 8, 24)$ using a 4×6 array, called the MOG, Conway [Co1] found an analogous description of $S(5, 6, 12)$ using a 3×4 array, called the MINIMOG. This section is devoted to the MINIMOG.

The *tetracode words* are

0	0	0	0	0	+	+	+	0	−	−	−
+	0	+	−	+	+	−	0	+	−	0	+
−	0	−	+	−	+	0	−	−	−	+	0

With "0" $= 0$, "+" $= 1$, "−" $= 2$, these vectors form a linear code over $GF(3)$. (This notation is Conway's. One must remember here that "+" + "+" = "−" and "−" + "−" = "+"!) They may also be described as the set of all 4-tuples in $GF(3)$ of the form

$$(0, a, a, a), \quad (1, a, b, c), \quad (2, c, b, a),$$

where abc is any cyclic permutation of 012.

The *MINIMOG in the shuffle labeling* is the 3×4 array

6	3	0	9
5	2	7	10
4	1	8	11

(For comparison, the MINIMOG in the modulo 11 labeling is given in Sect. 3.3.4 below. We shall use the shuffle labeling below unless stated otherwise.)

We label the rows as follows:

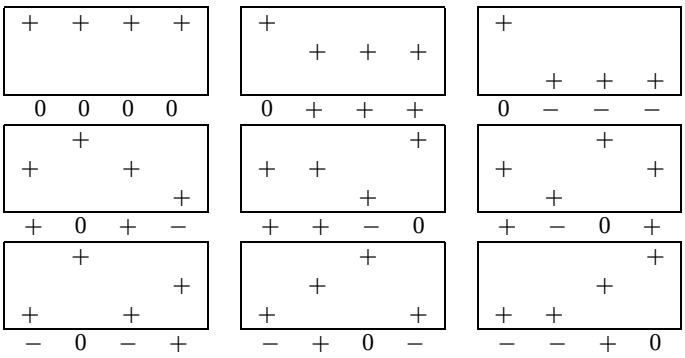
- the first row has label 1,
- the second row has label +,
- the third row has label −.

0	6	3	0	9
+	5	2	7	10
−	4	1	8	11

A *col* (or column) is a placement of three + signs in a column of the array:

+	+	+	+
+	+	+	+
+	+	+	+

A *tet* (or tetrad) is a placement of 4 "+" signs in the 3×4 array having entries corresponding to a tetracode as follows:



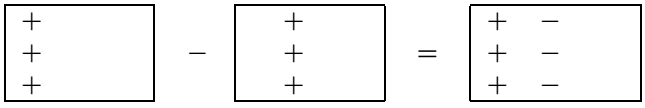
Each line in $GF(3)^2$ with finite slope occurs once in the 3×3 part of some tet. The *odd man out* for a column is the label of the row corresponding to the nonzero digit in that column; if the column has no nonzero digit, then the odd man out is a "?". Thus the tetracode words associated in this way to these patterns are the odd men out for the tets.

The *signed hexads* are the combinations 6-sets obtained from the MINIMOG from patterns of the form

$$\text{col} - \text{col}, \quad \text{col} + \text{tet}, \quad \text{tet} - \text{tet}, \quad \text{col} + \text{col} - \text{tet}.$$

Lemma 77 (Conway, [CS1], Chap. 11, p. 321) *If we ignore signs, then from these signed hexads we get the* 132 *hexads of a Steiner system* $S(5, 6, 12)$. *These are all possible* 6*-sets in the shuffle labeling for which the odd men out form a part* (*in the sense that and odd man out "?" is ignored or regarded as a "wild-card"*) *of a tetracode word and the column distribution is not* $0, 1, 2, 3$ *in any order.*[4]

Example 78 Associated to the col–col pattern



is the tetracode 0 0 ? ? and the signed hexad $\{-1, -2, -3, 4, 5, 6\}$ and the hexad $\{1, 2, 3, 4, 5, 6\}$. Indeed, if you identify $(\ , +, -)$ with $(0, 1, 2)$, then you can see this in SAGE as follows:

[4]That is to say, the following cannot occur: some column has 0 entries, some column has exactly 1 entry, some column has exactly 2 entries, and some column has exactly 3 entries.

SAGE

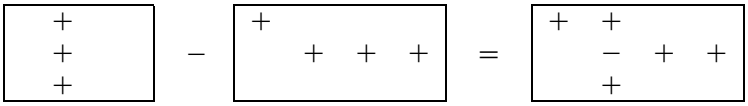
```
sage: M = Minimog(type="shuffle")
sage: M.minimog

[ 6  3  0  9]
[ 5  2  7 10]
[ 4  1  8 11]
sage: M.col[0] - M.col[1]

[1 2 0 0]
[1 2 0 0]
[1 2 0 0]
```

To verify that this is a hexad in the shuffle labeling, use the commands:

SAGE

```
sage: M.find_hexad([1,2,3,4,5])
     ([1, 2, 3, 4, 5, 6], ['square 8', 'picture 0'])
```

Associated to the col+tet pattern



is the tetracode $0\ +\ +\ +$ and the signed hexad $\{1, -2, 3, 6, 7, 10\}$ and the hexad $\{1, 2, 3, 6, 7, 10\}$. Likewise, you can check this in SAGE using the following commands:

SAGE

```
sage: M.col[1] - M.tet[1]

[1 1 0 0]
[0 2 1 1]
[0 1 0 0]
sage: M.find_hexad([1,2,3,6,7])
      ([1, 2, 3, 6, 7, 10], ['square 5', 'picture 0'])
```

Furthermore, it is known [Co1] that the Steiner system $S(5, 6, 12)$ in the shuffle labeling has the following properties.

- There are 11 hexads with total 21 and none with lower total.
- The complement of any of these 11 hexads in $\{0, 1, \ldots, 11\}$ is another hexad.
- There are 11 hexads with total 45 and none with higher total.

These facts will help us play the game mathematical blackjack in Sect. 3.3.2 below.

3.3.2 Construction of the Extended Ternary Golay Code

Definition 79 The *tetracode* is the $GF(3)$ code T with elements

$$(0,0,0,0),\quad (1,0,1,2),\quad (1,2,0,1),\quad (1,1,2,0),\quad (0,1,1,1),$$

$$(2,0,2,1),\quad (2,1,0,2),\quad (2,2,1,0),\quad (0,2,2,2).$$

It is a self-dual $(4,2,3)$ code.

Here is Conway's "tetracode" construction of the C_{12}. Represent each 12-tuple $c=(c_1,\ldots,c_{12})\in C_{12}$ as a 3×4 array

$$c=\begin{pmatrix} c_1 & c_2 & c_3 & c_4\\ c_5 & c_6 & c_7 & c_8\\ c_9 & c_{10} & c_{11} & c_{12}\end{pmatrix}.$$

The *projection* of c is

$$pr(c)=(c_5-c_9,c_6-c_{10},c_7-c_{11},c_8-c_{12}).$$

The *row score* is the sum of the elements in that row. The *column score* is the sum of the elements in that column. Of course, all computations are in $GF(3)$.

Lemma 80 (Conway) *An array c is in C_{12} if and only if*

- *the* (*common*) *score of all four columns equals the negative of the score of the top row*;
- $pr(c)\in T$.

There are several facts one can derive from this construction.

There are 264 codewords of weight 6, 440 codewords of weight 9, 24 codewords of weight 12, and codewords 729 total.

Pick an arbitrary subset of 9 elements taken from $\{1,2,3,\ldots,12\}$. It is the support of exactly two codewords of weight 9 in C_{12}. Pick a random subset S of 6 elements taken from $\{1,2,3,\ldots,12\}$. The probability that S is the support of some codeword of weight 6 in C_{12} is $1/7$.

Lemma 81 *For each weight* 6 *codeword c in C_{12}, there is a weight* 12 *codeword c' such that $c+c'$ has weight* 6.

If we call $c+c'$ a "complement" of c, then "the complement" is unique up to sign.

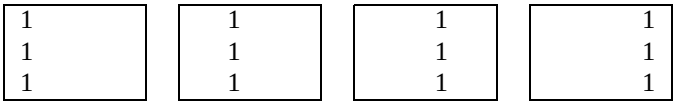
Proof The support of the codewords of weight 6 form a $S(5,6,12)$ Steiner system. Therefore, to any weight 6 codeword c, there is a codeword c'' whose support is in the complement of that of c. Let $c'=c''-c$. □

Remark 3 Although we shall not need it, it appears that for each weight 9 codeword c in C_{12}, there is a weight 12 codeword c' such that $c+c'$ has weight 6.

3.3.3 The "col/tet" Construction

In fact, this only constructs the codewords of weight 6, but since they generate the code, we can use them to compute a generating matrix for C_{12}.

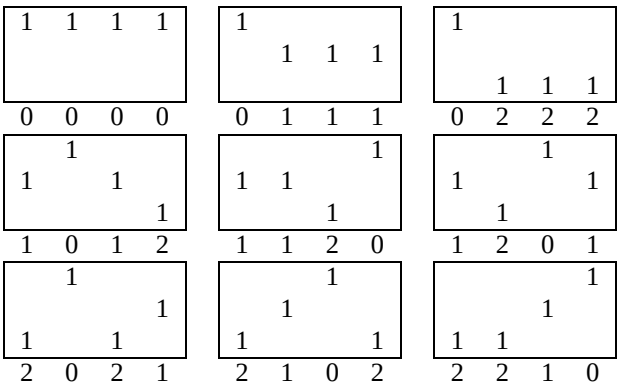
We translate the above definitions of col and tet using s slightly different notation (replacing each blank by 0, + by 1, and − by 2). A *col* (or column) is a placement of three 1s in a column of the array (a blank space represents a 0):



SAGE

```
sage: M = Minimog(type="shuffle")
sage: M.col[0]

[1 0 0 0]
[1 0 0 0]
[1 0 0 0]
sage: M.col[3]

[0 0 0 1]
[0 0 0 1]
[0 0 0 1]
```

A *tet* (or tetrad) is a placement of 4 1s having entries corresponding (as explained below) to a tetracode.



SAGE

```
sage: M = Minimog(type="shuffle")
sage: M.tet[0]
```

```
[1 1 1 1]
[0 0 0 0]
[0 0 0 0]
sage: M.tet[3]

[0 1 0 0]
[1 0 1 0]
[0 0 0 1]
sage: M.tet[8]

[0 0 0 1]
[0 0 1 0]
[1 1 0 0]
```

Each line in $\mathbb{F}_3^2$ with finite slope occurs once in the 3×3 part of some tet. Define the *label* of the first (top) row to be 0, of the second row to be -1, and of the bottom row to be 1. The *odd man out* for a column is the label of the row corresponding to the nonzero digit in that column; if the column has no nonzero digit, then the odd man out is a "?". Thus the tetracode words associated in this way to these patterns are the odd men out for the tets.

Example 82 Associated to the col–col pattern

$$\begin{array}{|c|}\hline 1\\ 1\\ 1\\ \hline\end{array} \quad - \quad \begin{array}{|c|}\hline \\ 1\\ 1\\ 1\\ \hline\end{array} \quad = \quad \begin{array}{|cc|}\hline 1 & 2\\ 1 & 2\\ 1 & 2\\ \hline\end{array}$$

is the tetracode $(0, 0, ?, ?)$.

Associated to the col+tet pattern

$$\begin{array}{|c|}\hline 1\\ 1\\ 1\\ \hline\end{array} \quad + \quad \begin{array}{|cccc|}\hline 1 & & & \\ & 1 & 1 & 1\\ \hline\end{array} \quad = \quad \begin{array}{|cccc|}\hline 1 & 1 & & \\ & 2 & 1 & 1\\ & 1 & & \\ \hline\end{array}$$

is the tetracode $(0, 1, 1, 1)$.

3.3.4 The Kitten Labeling

Recall that the MINIMOG in the shuffle labeling is the 3×4 array

$$\begin{array}{cccc} 6 & 3 & 0 & 9\\ 5 & 2 & 7 & 10\\ 4 & 1 & 8 & 11 \end{array}$$

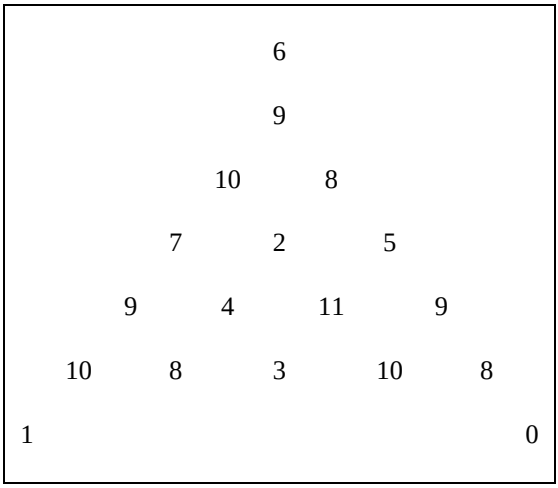


Fig. 3.2 The Kitten with shuffle labeling

In Conway, [Co1], the MINIMOG for the “modulo 11 labeling” is given by

$$\begin{array}{cccc} 0 & 3 & \infty & 2 \\ 5 & 9 & 8 & 10 \\ 4 & 1 & 6 & 7 \end{array}$$

Since Steiner systems $S(5, 6, 12)$ are unique up to relabelings, we should expect a “kitten” for the shuffle labeling. There is one, and this section describes it. Comparing this MINIMOG with that for the shuffle labeling, we obtain the following kitten in Fig. 3.2.

The “views” from each of the three “points at infinity” in the shuffle labeling is given in the following tables.

5	11	3
8	2	4
9	10	7

picture at 6

5	11	3
2	4	8
7	9	10

picture at 1

8	10	3
9	11	4
5	2	7

picture at 0

Example 83

- 0, 2, 4, 5, 6, 11 is a square in the picture at 1.
- 0, 2, 3, 4, 5, 7 is a cross in the picture at 0.

3.4 Playing “Mathematical Blackjack”

Mathematical blackjack is a two-person combinatorial game whose rules will be described below. What is remarkable about it is that a winning strategy, discovered by Conway and Ryba [CS2] and [KR], depends on knowing how to determine hexads in the Steiner system $S(5, 6, 12)$ using the shuffle labeling.

Winning ways in mathematical blackjack

Mathematical blackjack is played with 12 cards, labeled $0, \ldots, 11$ (for example: *king*, *ace*, 2, 3, ..., 10, *jack*, where the *king* is 0, and the *jack* is 11). Divide the 12 cards into two piles of 6 (to be fair, this should be done randomly). Each of the 6 cards of one of these piles are to be placed face up on the table. The remaining cards are in a stack which is shared and visible to both players. If the sum of the cards face up on the table is less than or equal to 21, then no legal move is possible, so you must shuffle the cards and deal a new game. (Conway [Co2] calls such a game $0 = \{|\}$; in this game the first player automatically loses, and so you courteously offered the first move!)

- Players alternate moves.
- A move consists of exchanging a card on the table with a lower card from the other pile.
- The player whose move makes the sum of the cards on the table under 21 loses.

The winning strategy (given below) for this game is due to Conway and Ryba [CS2], [KR]. There is a Steiner system $S(5, 6, 12)$ of hexads in the set $\{0, 1, \ldots, 11\}$. This Steiner system is associated to the MINIMOG of in the "shuffle numbering" rather than the "modulo 11 labeling."

Proposition 84 (Ryba) *For this Steiner system, the winning strategy is to choose a move which is a hexad from this system.*

This result is proven in [KR].

If you are unfortunate enough to be the first player starting with a hexad from $S(5, 6, 12)$, then, according to this strategy and properties of Steiner systems, there is no winning move. In a randomly dealt game there is a probability of

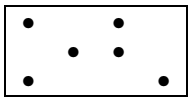
$$\frac{132}{\binom{12}{6}} = 1/7$$

that the first player will be dealt such a hexad, hence a losing position. In other words, we have the following result.

Lemma 85 *The probability that the first player has a win in mathematical blackjack* (*with a random initial deal*) *is* $6/7$.

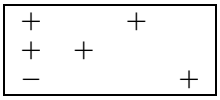
Example 86

- Initial deal: $0, 2, 4, 6, 7, 11$. The total is 30. The pattern for this deal is



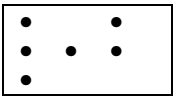
where • is a ±. No combinations of ± choices will yield a tetracode odd men out, so this deal is not a hexad.

- First player replaces 7 by 5: 0, 2, 4, 5, 6, 11. The total is now 28. (Note that this is a square in the picture at 1.) This corresponds to the col+tet



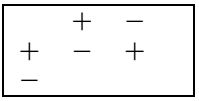
with tetracode odd men out − + 0 −.

- Second player replaces 11 by 7: 0, 2, 4, 5, 6, 7. The total is now 24. Interestingly, this 6-set corresponds to the pattern



(hence possible with tetracode odd men out 0 + + ?, for example). However, it has column distribution 3, 1, 2, 0, so it cannot be a hexad.

- First player replaces 6 by 3: 0, 2, 3, 4, 5, 7. (Note that this is a cross in the picture at 0.) This corresponds to the tet–tet pattern



with tetracode odd men out − − + 0. Cards total 21. First player wins.

Example 87 Actually, SAGE would play this game another way (also leading to a win).

SAGE

```
sage: M = Minimog(type="shuffle")
sage: M.blackjack_move([0,2,4,6,7,11])
'4 --> 3. The total went from 30 to 29.'
```

Is this really a hexad?

SAGE

```
sage: M.find_hexad([11,2,3,6,7])
 ([0, 2, 3, 6, 7, 11], ['square 9', 'picture 1'])
```

So, yes it is, but here is further confirmation:

SAGE

```
sage: M.blackjack_move([0,2,3,6,7,11])
This is a hexad.
There is no winning move, so make a random legal move.
[0, 2, 3, 6, 7, 11]
```

Yes, SAGE tells you that it is indeed a hexad.

Suppose that player 2 replaced the 11 by a 9. Your next move:

SAGE

```
sage:  M.blackjack_move([0,2,3,6,7,9])
'7 --> 1. The total went from 27 to 21.'
```

You have now won. SAGE will even tell you so:

SAGE

```
sage: M.blackjack_move([0,2,3,6,1,9])
 'No move possible. Shuffle the deck and redeal.'
```

3.5 Playing the Horses

How can a perfect, ternary code provide a good way of winning bets on horses? Suppose that 11 horses are racing and you must bet on the results of all 11 (each bet can be a win, a second place finish, or a third place finish). You wish to place the minimum number of bets such that, no matter what the results of the race, one of your bets picks the correct outcomes for at least 9 of the horses. The $[11, 6, 5]$ ternary Golay code corrects two errors and is perfect, so every vector (here, interpreted as a possible bet on the 11 horses) is at a Hamming distance of no more than 2 from some codeword. Therefore, if you bet the various combinations of wins, losses, and ties indicated by the $3^6 = 729$ codewords, one of your bets will have only 2 (or fewer) horse results predicted wrong.

How do you construct the $[11, 6, 5]$ ternary Golay code? Easy. Take the extended $[12, 6, 6]$ ternary Golay code and delete the last coordinate. In SAGE, it is very easy:

SAGE

```
sage: C = TernaryGolayCode()
sage: C
Linear code of length 11, dimension 6 over Finite Field of size 3
sage: C.random_element()
(1, 2, 1, 1, 2, 1, 1, 1, 2, 2, 2)
```

Typing `C.list()` will print all codewords, hence give a listing of all the bets you should make!

Chapter 4
The Riemann Hypothesis and Coding Theory

If the ring of integers $\mathbb{Z}$ is analogous with the polynomial ring $GF(p)[x]$, then we have the following comparisons:

$$\begin{array}{ccc} \mathbb{Z} & \leftrightarrow & GF(p)[x] \\ \text{prime numbers} & \leftrightarrow & \text{irreducible polynomials in } GF(p)[x], \end{array}$$

where p is a prime,

$$\begin{array}{ccc} \zeta(s) & \leftrightarrow & Z_{\mathbb{P}^1}(s) \\ \text{(Riemann zeta function)} & & \big(\text{Hasse–Weil zeta fcn of } \mathbb{P}^1/GF(p)\big). \end{array}$$

This analogy extends (no pun intended) to finite algebraic extensions, leading to analogies between the Dedekind zeta function $\zeta_K(s)$ of a number field K and the Hasse–Weil zeta function of a smooth projective curve defined over a finite field. If the ring of integers $\mathcal{O}$ of a number field K are analogous with the coordinate ring $GF(q)(X)$ of a smooth projective curve $X/GF(q)$, then we have the following comparisons:

$$\begin{array}{ccc} \mathcal{O} & \leftrightarrow & GF(q)(X) \\ \text{prime ideals in } \mathcal{O} & \leftrightarrow & \text{prime ideals in } GF(q)(X), \end{array}$$

where q is a prime power (discussed briefly using SAGE in Sect. 4.4.4 below),

$$\begin{array}{ccc} \zeta_K(s) & \leftrightarrow & Z_X(s) \\ \text{(Dedekind zeta function)} & & \big(\text{Hasse–Weil zeta fcn of } X/GF(q)\big). \end{array}$$

The basic idea behind this is that if we believe that the Riemann hypothesis holds for the Riemann zeta function, and its analogs for Dedekind zeta functions, then we should also believe in its truth for the Hasse–Weil zeta function for curves. (The Riemann hypothesis for curves was settled by A. Weil in the 1940s.)

D. Joyner, J.-L. Kim, *Selected Unsolved Problems in Coding Theory*,
Applied and Numerical Harmonic Analysis,
DOI 10.1007/978-0-8176-8256-9_4,

I. Duursma [D1, D2, D3, D4, D5, D6] has defined a zeta function for linear codes and has extended this analogy to linear codes,[1] so that in some vague sense:

Hasse-Weil zeta function of a curve $\Leftrightarrow$ Duursma zeta function of a code.

In particular, there is an analog of the well-known Riemann hypthesis in coding theory. This chapter is devoted to explaining the fascinating details surrounding this open question.

4.1 Introduction to the Riemann Zeta Function

The *Riemann hypothesis*, first formulated by Bernhard Riemann in 1859, is one of the most famous and important unsolved problems in mathematics. The Riemann hypothesis is a conjecture about the distribution of the zeros of the Riemann zeta-function $\zeta(s)$. The Riemann zeta-function $\zeta(s)$ is the function of a complex variable s initially defined by the following infinite series:

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}$$

for values of s with real part greater than one. A globally convergent series for the zeta function, valid for all complex numbers s except $s = 1$, was conjectured by Konrad Knopp and proved by Helmut Hasse in 1930:

$$\zeta(s) = \frac{1}{1 - 2^{1-s}} \sum_{n=0}^{\infty} \frac{1}{2^{n+1}} \sum_{k=0}^{n} (-1)^k \binom{n}{k} (k+1)^{-s}. \tag{4.1.1}$$

Another interesting property that the Riemann zeta function has is its so-called functional equation:

$$\zeta(s) = 2^s \pi^{s-1} \sin\left(\frac{\pi s}{2}\right) \Gamma(1-s)\zeta(1-s).$$

If we let

$$\xi(s) = \pi^{-s/2} \Gamma\left(\frac{s}{2}\right) \zeta(s),$$

then we can rewrite this as

$$\xi(1-s) = \xi(s).$$

Because of (4.1.1), the Riemann zeta-function is defined for all complex numbers $s \neq 1$. It has zeros at the negative even integers (i.e., at $s = -2, -4, -6, \ldots$). These

[1] In fact, this analogy can be extended to an analogy between curves and matroids. The analogy with curves and codes is discussed in more detail in Sect. 4.4.4 below.

are called the *trivial zeros*. The Riemann hypothesis is concerned with the *nontrivial zeros* and states that:

The real part of any nontrivial zero of the Riemann zeta function is $\frac{1}{2}$.

It has been an open question for almost 150 years, despite attracting concentrated efforts from many outstanding mathematicians.

4.2 Introduction to the Duursma Zeta Function

Let C be an $[n, k, d]_q$ code, i.e., a linear code over $GF(q)$ of length n, dimension k, and minimum distance d. Recall that the Singleton bound states that $d + k \leq n + 1$ and that codes which satisfy equality in this bound are called MDS (maximum distance separable) codes.

Motivated by analogies with local class field theory, in [D1] Iwan Duursma introduced the *zeta function* $Z = Z_C$ associated to a linear code C over a finite field,

$$Z(T) = \frac{P(T)}{(1-T)(1-qT)}, \tag{4.2.1}$$

where $P(T) = P_C(T)$ is a polynomial of degree $n + 2 - d - d^\perp$, called the *zeta polynomial*.[2] If C is self-dual (i.e., $C = C^\perp$), it satisfies a functional equation of the form

$$P(t) = q^g t^{2g} P\left(\frac{1}{qt}\right).$$

This does not look too much like the functional equation for the Riemann zeta function (yet).

If $\gamma = \gamma(C) = n + 1 - k - d$ (the *genus* of C) and if

$$z_C(T) = Z_C(T)T^{1-\gamma},$$

then the functional equation can be written in the form

$$z_{C^\perp}(T) = z_C(1/qT).$$

If we let

$$\zeta_C(s) = Z_C\left(q^{-s}\right)$$

and

$$\xi_C(s) = z_C\left(q^{-s}\right),$$

[2] In general, if C is an $[n, k, d]$-code, then we use $[n, k^\perp, d^\perp]$ for the parameters of the dual code, $C^\perp$. It is a consequence of Singleton's bound that $n + 2 - d - d^\perp \geq 0$, with equality when C is an MDS code.

then ζ_C and ξ_C have the same zeros, but ξ_C is "more symmetric" since the functional equation expressed in terms of it becomes[3]

$$\xi_{C^\perp}(s) = \xi_C(1-s).$$

Abusing terminology, we call both Z_C and ζ_C the *Duursma zeta function* of C.

4.3 Introduction

Recall that a linear code C is called an $[n,k,d]_q$-code if it is a k-dimensional subspace of $GF(q)^n$ having minimum distance d,

$$d = \min_{c\in C, c\neq 0} \mathrm{wt}(c),$$

where wt is the Hamming weight of a codeword. The dual code of C, denoted $C^\perp$, has parameters $[n, n-k, d^\perp]$ for some $d^\perp \geq 1$. The *genus* of an $[n,k,d]_q$-code C is defined by

$$\gamma(C) = n+1-k-d.$$

This measures how "far away the code is from being MDS." If C is an algebraic-geometric code constructed from the Riemann–Roch space of an algebraic curve over $GF(q)$, then it is often equal to the genus of the curve (see [TV] for details).

Note that if C is a self-dual code, then its genus satisfies $\gamma = n/2+1-d$.

4.3.1 Virtual Weight Enumerators

The following definition generalizes the notion introduced in Sect. 2.1 above.

Definition 88 A homogeneous polynomial $F(x,y) = x^n + \sum_{i=1}^n f_i x^{n-i} y^i$ of degree n with complex coefficients is called a *virtual weight enumerator* with *support* $\mathrm{supp}(F) = \{0\} \cup \{i \mid f_i \neq 0\}$. If $F(x,y) = x^n + \sum_{i=d}^n A_i x^{n-i} y^i$ with $A_d \neq 0$, then we call n the *length* of F and d the *minimum distance* of F. Such an F of even degree satisfying (2.2.1) is called a *virtually self-dual weight enumerator over* $GF(q)$ having *genus*

$$\gamma(F) = n/2+1-d.$$

If $b > 1$ is an integer and $\mathrm{supp}(F) \subset b\mathbb{Z}$, then the virtual weight enumerator F is called *b-divisible*.

[3]This notation is inspired by analogous notation used for functions associated with the classical Riemann zeta function. See any book on the Riemann zeta function or http://en.wikipedia.org/wiki/Riemann_zeta_function.

The classification of nontrivial formally self-dual divisible codes into the four Types (as defined in Chap. 2) has a virtually self-dual weight enumerator analog. In other words, the Gleason–Pierce theorem has a strengthening where the hypothesis does not require the existence of a code, only a form which certain invariance properties.

Theorem 89 (Gleason–Pierce–Assmus–Mattson) *Let F be a b-divisible virtually self-dual weight enumerator over $GF(q)$.*

Then either

I. $q = b = 2$,
II. $q = 2, b = 4$,
III. $q = b = 3$,
IV. $q = 4, b = 2$,
V. *q is arbitrary,* $b = 2$, *and* $F(x, y) = (x^2 + (q-1)y^2)^{n/2}$.

Proof The proof (or proofs—there are now two of them) is due to Assmus and Mattson. The easiest place to access the argument is in the excellent survey paper by Sloane [Sl]. The rough idea is as follows (for details, please see Sect. 6.1 in Sloane's paper).

Let G denote the subgroup of $GL(2, \mathbb{C})$ generated by the matrix of the "MacWilliams transform"

$$F(x, y) \longmapsto F\left(\frac{x + (q-1)y}{\sqrt{q}}, \frac{x - y}{\sqrt{q}}\right)$$

together with the diagonal matrices having bth roots of unity on the diagonal (since $F(x, y) \longmapsto F(\zeta x, y)$ and $F(x, y) \longmapsto F(x, \zeta y)$ both fix F if $\zeta \in F$ is any bth root of unity). Let G' denote its image in $PGL(2, \mathbb{C})$. Think of $F(x, y)$ as a function $f(z)$ of $z = x/y$ on $\mathbb{P}^1$. Let m denote the number of zeros of f (not counting multiplicity). By the invariance properties, $m = 1$ is impossible. If $m = 2$, then the invariance property implies (V). If $m \geq 3$, then G' must be finite. The classification of finite subgroups of $PGL(2, \mathbb{C})$ results in the remaining possibilities (I), ..., (IV). □

Next we give the virtual weight enumerator analog of Definition 39 above.

Definition 90

- Let $F(x, y)$ be a virtually self-dual weight enumerator. If $b > 1$ is an integer and $\mathrm{supp}(F) \subset b\mathbb{Z}$, then F is called *b-divisible*.
- If F is a b-divisible virtually self-dual weight enumerator over $GF(q)$, then F is called

$$\begin{cases} \text{Type I} & \text{if } q = b = 2,\ 2|n, \\ \text{Type II} & \text{if } q = 2,\ b = 4,\ 8|n, \\ \text{Type III} & \text{if } q = b = 3,\ 4|n, \\ \text{Type IV} & \text{if } q = 4,\ b = 2,\ 2|n. \end{cases}$$

Theorem 91 (Sloane–Mallows–Duursma) *If F is a b-divisible virtually self-dual weight enumerator with length n and minimum distance d, then*

$$d \leq \begin{cases} b[\frac{n}{b(b+1)}] + b & \text{if } F \text{ is Type 1,} \\ b[\frac{n}{b(b+2)}] + b & \text{if } F \text{ is Type 2.} \end{cases} \tag{4.3.1}$$

In particular,

$$d \leq \begin{cases} 2[n/8] + 2 & \text{if } F \text{ is Type I,} \\ 4[n/24] + 4 & \text{if } F \text{ is Type II,} \\ 3[n/12] + 3 & \text{if } F \text{ is Type III,} \\ 2[n/6] + 2 & \text{if } F \text{ is Type IV.} \end{cases}$$

Proof This is only stated for self-dual codes, but the proof of Theorem 1 and the argument in Sect. 1.1 of Duursma [D3] hold more generally for virtually self-dual weight enumerators. A complete proof is given in Appendix 7.4 below. □

Definition 92 A virtually self-dual weight enumerator F is called *extremal* if the bound in Theorem 91 holds with equality.

Remark 8

- Here is a more general definition. Let G be a subgroup of $GL(2, \mathbb{C})$ containing $\sigma = \frac{1}{\sqrt{q}}\left(\begin{smallmatrix} 1 & q-1 \\ 1 & -1 \end{smallmatrix}\right)$, acting on $\mathbb{C}[x, y]$ by $\sigma : F(x, y) \longmapsto F(\sigma(x, y)^t)$, and $\chi : G \to \mathbb{C}^\times$ a character. Call a virtual weight enumerator F of length n a *formally χ-self-dual weight enumerator*, or a *virtually self-dual weight enumerator twisted by χ*, if[4]

$$F(x, y) = \chi(\sigma) F\left(\frac{x + (q-1)y}{\sqrt{q}}, \frac{x - y}{\sqrt{q}}\right).$$

  The virtually self-dual weight enumerator definition above is the special case where χ is a trivial. This "twisted" definition also covers, for example, the case of Ozeki's "formal weight enumerators" in [O]. For brevity, we call F a *twisted virtually self-dual weight enumerator* if it satisfies

$$F(x, y) = -F\left(\frac{x + (q-1)y}{\sqrt{q}}, \frac{x - y}{\sqrt{q}}\right). \tag{4.3.2}$$

  Much of the theory of zeta functions for virtually self-dual weight enumerators also applies to twisted virtually self-dual weight enumerators. See Chinen [Ch1, Ch2] and Sect. 4.8 below.

[4]This "twisted" terminology is motivated by terminology in automorphic forms and arithmetical algebraic geometry for analogous objects.

- Note that a virtual weight enumerator does not depend on a prime power q but a virtually self-dual weight enumerator does depend on q through (2.2.1).

Definition 93 A virtual weight enumerator F is formally identified with an object we call a *virtual code* C subject only to the following condition: we formally extend the definition of $C \longmapsto A_C$ to all virtual codes by $A_C = F$. Of course, if F is the weight enumerator of an actual code, say C', then we have $A_C = F = A_{C'}$. In other words, a virtual code is only well defined up to formal equivalence. If C_1 and C_2 are virtual codes, then we define $C_1 + C_2$ to be the virtual code associated to the virtual weight enumerator $A_{C_1}(x, y) + A_{C_2}(x, y)$.

The following question is really more a question of the classification of self-dual codes than of virtually self-dual weight enumerators. An excellent reference is the book [NRS].

Open Problem 18 Given a virtually self-dual weight enumerator F, find necessary and sufficient conditions (short of enumeration) which determine whether or not F arises as the weight enumerator of some self-dual code C.

4.4 The Zeta Polynomial

We shall give three definitions of the zeta polynomial, all due to Duursma.

4.4.1 First Definition

Definition 94 A polynomial $P(T)$ for which

$$\frac{(xT + (1-T)y)^n}{(1-T)(1-qT)} P(T) = \cdots + \frac{A_C(x, y) - x^n}{q-1} T^{n-d} + \cdots$$

is called a *Duursma zeta polynomial of* C.

The *Duursma zeta function* is defined in terms of the zeta polynomial by means of (4.2.1) above.

Lemma 95 *If we expand* $\frac{(xT+y(1-T))^n}{(1-T)(1-qT)}$ *in powers of* T, *we find that it is equal to*

$$b_{0,0}y^n T^0 + \left(b_{1,1}xy^{n-1} + b_{1,0}y^n\right)T^1 + \left(b_{2,2}x^2y^{n-2} + b_{2,1}xy^{n-1} + b_{2,0}y^n\right)T^2 + \cdots + \left(b_{n-d,n-d}x^{n-d}y^d + b_{n-d,n-d-1}x^{n-d-1}y^{d+1} + \cdots + b_{n-d,0}y^n\right)T^{n-d} + \cdots,$$

where $b_{i,j}$ are the coefficients given by

$$b_{k,\ell} = \sum_{i=\ell}^{k} \frac{q^{k-i+1}-1}{q-1} \binom{n}{i} \binom{i}{\ell}$$

for $0 \le \ell \le k \le n-d$ and $b_{k,\ell} = 0$ otherwise.

Proof Use (4.4.2) below and compare the coefficients. □

Proposition 96 *The Duursma zeta polynomial $P = P_C$ exists and is unique, provided that $d^\perp \ge 2$.*

Proof This is proven in the appendix to Chinen [Ch2]. Here is the rough idea. Expand $\frac{(xT+y(1-T))^n}{(1-T)(1-qT)}$ in powers of T, as in Lemma 95 above. The Duursma polynomial is a polynomial of degree $n+2-d-d^\perp$. Provided that $d^\perp \ge 2$, the Duursma polynomial can be written as $P(T) = a_0 + a_1 T + \cdots + a_{n-d}T^{n-d}$. Now, use

$$\frac{(xT+y(1-T))^n}{(1-T)(1-qT)} P(T) = \cdots + \frac{F(x,y)-x^n}{q-1} T^{n-d} + \cdots$$

to express the coefficients by means of the matrix equation $B \cdot \vec{a} = \vec{A}$ given by

$$\begin{pmatrix} b_{0,0} & b_{1,0} & \dots & b_{n-d,0} \\ 0 & b_{1,1} & \dots & b_{n-d,1} \\ 0 & 0 & b_{2,2} & \cdots \\ \vdots & & \ddots & \vdots \\ 0 & \dots & 0 & b_{n-d,n-d} \end{pmatrix} \begin{pmatrix} a_{n-d} \\ a_{n-d-1} \\ \vdots \\ a_0 \end{pmatrix} = \begin{pmatrix} A_n/(q-1) \\ A_{n-1}/(q-1) \\ \vdots \\ A_d/(q-1) \end{pmatrix}. \tag{4.4.1}$$

Thanks to Lemma 95 above, we know that the diagonal entries of this matrix are binomial coefficients, $b_{i,i} = \binom{n}{i}$, hence are nonzero. Therefore the matrix is invertible, and the existence is established. □

Here is a corollary of the proof. (These identities are given in [D5] as (5) and (6); see also (4.1) of [D4].)

Corollary 97 (Duursma) *If $d^\perp \ge 2$, then $P(0) = (q-1)^{-1}\binom{n}{d}^{-1} A_d$, and*

$$\frac{A_{d+1}}{q-1} = \binom{n}{d+1} \big(P(0)(q-d) + P'(0)\big).$$

In particular, P always has a nonzero positive constant coefficient.

Proof By the above proof, $b_{n-i,n-i}$ is the ith binomial coefficient, and so the first equation follows from the system of equations in (4.4.1).

The second equation follows similarly, so its proof is omitted. □

Example 98 Consider the self-dual code C of length $n = 6$, dimension $k = 3$, and minimum distance $d = 2$. This is unique up to equivalence and has weight enumerator $W(x, y) = x^6 + 3x^4y^2 + 3x^2y^4 + y^6$. The SAGE commands

SAGE

```
sage: q,T,x,y = var("q,T,x,y")
sage: f1 = lambda q,T,N:
                sum([ sum([q^i for i in range(k+1)])*T^k for k in range(N)])
sage: f2 = lambda x,y,T,n:
                sum([ binomial(n,j)*(x-y)^j*y^(n-j)*T^j for j in range(n+1)])
sage: a0,a1,a2,a3,a4 = var("a0,a1,a2,a3,a4")
sage: F = expand(f1(2,T,6)*f2(x,y,T,6)*(a0+a1*T+a2*T^2+a3*T^3+a4*T^4))
```

compute the first 6 terms (as a power series in T) of the series $\frac{(xT+y(1-T))^n}{(1-T)(1-qT)}P(T)$ when $q = 2$, $n = 6$, $k = 3$, and $d = 2$. Next, we compute the coefficients and read off the matrix B:

SAGE

```
sage: aa = (F.coeff("T^4")).coeffs("x")
sage: v = [expand(aa[i][0]/y^(6-i)) for i in range(5)]
sage: B0 = [v[0].coeff("a%s"%str(i)) for i in range(5)]
sage: B1 = [v[1].coeff("a%s"%str(i)) for i in range(5)]
sage: B2 = [v[2].coeff("a%s"%str(i)) for i in range(5)]
sage: B3 = [v[3].coeff("a%s"%str(i)) for i in range(5)]
sage: B4 = [v[4].coeff("a%s"%str(i)) for i in range(5)]
sage: B0.reverse(); B1.reverse(); B2.reverse(); B3.reverse(); B4.reverse()
sage: B = matrix([B0,B1,B2,B3,B4])
sage: B

[ 1 -3 4 -2 1]
[ 0 6 -12 12 0]
[ 0 0 15 -15 15]
[ 0 0 0 20 0]
[ 0 0 0 0 15]
```

Note that the diagonal entries are binomial coefficients.

Finally, we compute the vector $\vec{A}$ and solve the equation $B \cdot \vec{a} = \vec{A}$:

SAGE

```
sage: Wmx6 = 3*x^4*y^2+3*x^2*y^4+y^6
sage: c = [Wmx6(1,y).coeff("y^%s"%str(i)) for i in range(2,7)]
sage: c.reverse()
sage: A = vector(c)
sage: (B^(-1)*A).list()
[4/5, 0, 0, 0, 1/5]
```

This implies that the zeta function of C is given by $P(T) = \frac{1}{5} + \frac{4}{5}T^4$.

Duursma has given several definitions (all equivalent of course) of $P(T)$. Before stating another one, we need the following definition and lemma.

Definition 99 Define c_j by

$$\frac{(xT + (1-T)y)^n}{(1-T)(1-qT)} = \sum_{k=0}^{\infty} c_k(x, y)T^k.$$

Define $M_{n,\delta}$ by

$$M_{n,\delta}(x, y) = x^n + (q-1)c_{n-\delta}(x, y).$$

This is called the *MDS virtual weight enumerator of length n and distance δ*.

It is not hard to see that

$$\frac{1}{(1-T)(1-qT)} = \sum_{j=0}^{\infty} \frac{q^{j+1}-1}{q-1} T^j$$

and of course

$$\big(xT + (1-T)y\big)^n = \sum_{i=0}^{n} \binom{n}{i} y^{n-i}(x-y)^i T^i.$$

Therefore,

$$c_k(x, y) = \sum_{i+j=k} \frac{q^{j+1}-1}{q-1} \binom{n}{i} y^{n-i}(x-y)^i. \tag{4.4.2}$$

A version of the following result is stated in Duursma [D5] (see his (9)).

Lemma 100 *If F is a virtual weight enumerator of length n and minimum distance d, then there are coefficients $c \in \mathbb{Q}$ and $a_i = a_j(F) \in \mathbb{Q}$ such that*

$$F(x, y) = cx^n + a_0 M_{n,d}(x, y) + a_1 M_{n,d+1}(x, y) + \cdots + a_r M_{n,d+r}(x, y) \tag{4.4.3}$$

for some r, $0 \le r \le n-d$. In fact, $c = 1 - a_0 - \cdots - a_r$.

Proof The functions $M_{n,d+i}(x, y) - x^n$ form a basis for the vector space $V = \{\sum_{i=d}^{n} b_i x^{n-i} y^i \mid b_i \in \mathbb{Q}\}$.

Consider the equation

$$\begin{aligned} F(x, y) - x^n = a_0\big(M_{n,d}(x, y) - x^n\big) + a_1\big(M_{n,d+1}(x, y) - x^n\big) \\ + \cdots + a_r\big(M_{n,d+r}(x, y) - x^n\big). \end{aligned}$$

If $r = \dim(V) - 1$, then one can solve for the $a_0, \ldots, a_r$. Without loss of generality, we may take $r \ge 0$ to be as small as possible. We have then

$$\begin{aligned} F(x, y) = (1 - a_0 - \cdots - a_r)x^n + a_0 M_{n,d}(x, y) + a_1 M_{n,d+1}(x, y) + \cdots \\ + a_r M_{n,d+r}(x, y). \end{aligned}$$

□

Example 101 We use SAGE [S] to compute examples.

When $q = 2$,

$$M_{10,5}(x, y) = -34y^{10} + 220xy^9 - 585x^2y^8 + 840x^3y^7 - 630x^4y^6 + 252x^5y^5 + x^{10},$$

and when $q = 3$,

$$M_{12,5}(x, y) = -48y^{12} + 1152xy^{11} - 2376x^2y^{10} + 8360x^3y^9 - 7920x^4y^8 + 9504x^5y^7 - 3696x^6y^6 + 1584x^7y^5 + x^{12}.$$

The negative coefficients in these polynomials are consistent with the fact that for codes of dimension greater than 1, the length of an MDS code satisfies the bound $n \leq q + k - 1$ (see, for example, pages 12–13 in [TV]). In the first example, a $[10, 6, 5]_2$ code must satisfy $10 \leq 2 + 6 - 1$ (so it does not exist), and, in the second example, a $[12, 8, 5]_3$ code must satisfy $12 \leq 3 + 8 - 1$ (so it does not exist).

On the other hand, when $q = 13$,

$$M_{12,5}(x, y) = 312177312y^{12} + 312178752xy^{11} + 143076384x^2y^{10} + 39755760x^3y^9 + 7436880x^4y^8 + 1007424x^5y^7 + 88704x^6y^6 + 9504x^7y^5 + x^{12}.$$

Indeed, according to SAGE's `ReedSolomonCode` command, there is an MDS code C having parameters $[12, 8, 5]_{13}$:

SAGE

```
sage: C = ReedSolomonCode(12,8,GF(13))
sage: C.spectrum()

[1,
 0,
 0,
 0,
 0,
 9504,
 88704,
 1007424,
 7436880,
 39755760,
 143076384,
 312178752,
 312177312]
```

This SAGE session tells us that

$$\text{spec}(C) = [1, 0, 0, 0, 0, 9504, 88704, 1007424, 7436880, 39755760, 143076384, 312178752, 312177312],$$

as the above (independently obtained) computation implies.

These virtual weight enumerators are computed using the following SAGE code:

SAGE

```
sage: R = PolynomialRing(QQ,2,"xy")
sage: x,y = R.gens()
sage: f = lambda q,n,m :\
 (x*T+y*(1-T))^(n)*sum([T^i for i in range(m)])\
 *sum([(q*T)^i for i in range(m)])
sage: M = lambda q,n,d,m : (f(q,n,m).list())[d]*(q-1)+x^n
```

As long as `m` is taken to be sufficiently large, this code will return the correct value of $M_{n,d}$.

Example 102 The Duursma zeta function of the $[2^r-1, 2^r-r-1, 3]$-Hamming code, $\mathrm{Ham}(r, GF(2))$, can be computed using the following SAGE commands:

SAGE

```
sage: C = HammingCode(3,GF(2))
sage: C.zeta_function()
 (2/5*T^2 + 2/5*T + 1/5)/(2*T^2 - 3*T + 1)
sage: C = HammingCode(4,GF(2))
sage: C.zeta_function()
(16/429*T^6 + 16/143*T^5 + 80/429*T^4 + 32/143*T^3\
+ 30/143*T^2 + 2/13*T + 1/13)/(2*T^2 - 3*T + 1)
```

In other words,

$$Z_{\mathrm{Ham}(3,GF(2))}(T) = \frac{\frac{1}{5}(2T^2+2T+1)}{2T^2-3T+1},$$

and

$$Z_{\mathrm{Ham}(4,GF(2))}(T) = \frac{\frac{1}{429}(16T^6+48T^5+80T^4+96T^3+90T^2+66T+33)}{2T^2-3T+1}.$$

Example 103 The Duursma zeta function of the maximal binary linear self-dual doubly even code of length 8 can be computed using the following different SAGE commands:

SAGE

```
sage: MS = MatrixSpace(GF(2),4,8)
sage: G =
MS([[1,1,1,1,0,0,0,0],[0,0,1,1,1,1,0,0],[0,0,0,0,1,1,1,1],
   [1,0,1,0,1,0,1,0]])
sage: C = LinearCode(G)
sage: C
Linear code of length 8, dimension 4 over Finite Field of size 2
sage: C.zeta_function()
```

```
(2/5*T^2 + 2/5*T + 1/5)/(2*T^2 - 3*T + 1)
sage: C.sd_zeta_polynomial()
2/5*T^2 + 2/5*T + 1/5
sage: C == C.dual_code()
True
```

In other words,

$$P_C(T) = \left(2T^2 + 2T + 1\right)/5.$$

4.4.2 Second Definition

Here is Duursma's second definition of the zeta polynomial.

Definition 104 Let $F = A_C$ denote the weight enumerator of a $[n, k, d]_q$-code C. Using the coefficients $a_j = a_j(F)$ of (4.4.3), define

$$P(T) = P_C(T) = a_0 + a_1 T + \cdots + a_r T^r.$$

This $P(T)$ is the *Duursma zeta polynomial of* C.

More generally, if F is an virtual weight enumerator and the coefficients $a_j = a_j(F)$ are as in (4.4.3), define $P(T) = P_F(T) = a_0 + a_1 T + \cdots + a_r T^r$.

Note that by comparing coefficients of x^n on both sides of (4.4.3), we see that $a_0 + \cdots + a_r = 1$ is equivalent to $P(1) = 1$.

Example 105 Note that if C is an MDS code of length n and minimum distance d over $GF(q)$, then $A_C = M_{n,d}$ (this is proven as part of the discussion in Sect. 2 of Duursma [D2]). This forces $c = 0$, $a_0 = 1$ in (4.4.3), so[5] $P(t) = 1$.

Remark 9 Note that $[n, k, d]$ makes sense as parameters of a virtual weight enumerator when F is a weight enumerator of an actual code C (so $F = A_C$) or when F is a virtually self-dual weight enumerator (so $\gamma = n/2 - d + 1$, where n and d are as in Definition 88) or a virtual MDS code (so $k = n + 1 - d$).

Lemma 106 *The Duursma zeta function of Definition 94 is the same as the Duursma zeta function of Definition 104.*

Proof By Definition 99, the zeta polynomial of Definition 94 associated to $F = A_C$ is T^r if you replace $F = A_C$ by $F = M_{n,d+j}$:

$$\frac{(xT + (1-T)y)^n}{(1-T)(1-qT)} T^j = \cdots + \frac{M_{n,d+j}(x, y) - x^n}{q-1} T^{n-d} + \cdots.$$

[5] See also Duursma's Proposition 1 in [D5] and Chinen's Theorem 3.2 in [Ch3].

Multiply by a_j and sum both sides over $j \in \{0, \ldots, r\}$ to obtain Definition 104. Therefore, $P(T)$ satisfying Definition 94 also satisfies Definition 104. □

4.4.3 Third Definition

In preparation for the third definition, which originated in Sect. 7 of Duursma [D1], we introduce some notation.

Let C be an $[n, k, d]_q$ code, let $S \subset \{1, 2, \ldots, n\}$ be a subset, let C_S denote the subcode of C of codewords with support contained in S, and let $k_S = k_S(C)$ denote the dimension of C_S.

Lemma 107 *The dimension k_S satisfies*

$$k_S = \begin{cases} 0 & \text{for } 0 \le |S| < d, \\ k - (n - |S|) & \text{for } n - d^\perp < |S| \le n. \end{cases}$$

When $d \le |S| \le n - d^\perp$, then k_S depends on S and C in a more subtle way.

Proof It follows from the definition of the minimum distance d that $k_S = 0$ if $0 \le |S| < d$. If C is $[n, k, d]$, then the dual code $C^\perp$ is $[n, n-k, d^\perp]$, so $n - k + d^\perp \le n + 1$, or $d^\perp \le k + 1$. If $S^c = \{j \mid 1 \le j \le n, j \notin S\}$, then C_S is isomorphic to the code "shortened on S^c." The dimensions of such shortened codes are given in Theorem 1.5.7 in [HP1]. In particular, if $|S^c| < d^\perp$, then we find $k_S = n - |S^c| - (n - k) = k - |S^c|$, as desired. □

The *binomial moments* of C are the integers $B_0^1, B_1^1, B_2^1, \ldots$ defined by

$$B_i^1 = B_i^1(C) = \sum_{\substack{S \\ |S|=i}} \frac{q^{k_S} - 1}{q - 1}.$$

Lemma 108 *The binomial moments satisfy*

$$B_i^1 = \begin{cases} 0 & \text{for } 0 \le i < d, \\ \binom{n}{i} \frac{q^{i+k-n} - 1}{q-1} & \text{for } n - d^\perp < i \le n. \end{cases}$$

Proof This is an easy corollary of the above lemma. □

The numbers

$$b_i = b_i(C) = B_{d+i}^1 \Big/ \binom{n}{d+i} \tag{4.4.4}$$

are called the *normalized binomial moments* of C ($0 \le i \le n-d$). We extend this to all $i \in \mathbb{Z}$ by

$$b_i = b_i(C) = \begin{cases} 0 & \text{for } i < 0, \\ \frac{q^{i+d+k-n}-1}{q-1} & \text{for } n - d^\perp - d < i. \end{cases}$$

Finally, we can give Duursma's third definition.

Definition 109 Define the *zeta function* of C to be the generating function of the normalized binomial moments of the code:

$$Z(T) = \sum_{i=0}^{\infty} b_i T^i.$$

This is a rational function (see Duursma [D1], Sect. 7),

$$Z(T) = \frac{P(T)}{(1-T)(1-qT)},$$

where

$$P(T) = a_0 + a_1 T + \cdots + a_{n+2-d-d^\perp} T^{n+2-d-d^\perp}$$

is the zeta polynomial, and

$$a_i = b_i - (q+1)b_{i-1} + q b_{i-2}. \tag{4.4.5}$$

Lemma 110 *The Duursma zeta function of Definition 109 is the same as the Duursma zeta function of Definition 94.*

Proof If

$$B^1(x, y) = \sum_{j=0}^{n} B_j^1 x^{n-j} y^j$$

and $A_C(x, y) = x^n + (q-1)A^1(x, y)$, then it is known[6] that $B^1(x, y) = A^1(x + y, y)$. Therefore, $\frac{A_C(x,y)-x^n}{q-1} = B^1(x - y, y)$ and

$$(zT + y)^n Z(T) = \cdots + B^1(z, y)T^{n-d} + \cdots$$

(where $z = x - y$) defines the Duursma zeta polynomial of C in the sense of Definition 94. Let us compare coefficients of $z^\ell T^{n-d}$ on both sides. On the right-hand side, it is $B^1_{n-\ell}$, and on the other side, it is $\binom{n}{\ell} b_{n-d-\ell}$. We must verify that these are

[6] This is proven in Sect. 9 of [D5]. See Theorem 1.1.26 and Exercise 1.1.27 in [TV] for a closely related result.

the same. However, this is the formula for the normalized binomial moment and so is, by definition, true. □

As a corollary, we find that if the weight enumerator A_C is known, then

$$B^1(x, y) = \frac{A_C(x + y, y) - (x + y)^n}{q - 1} = \sum_{j=0}^{n} B_j^1 x^{n-j} y^j$$

is easy to compute, and the coefficients of the zeta polynomial are given by (4.4.4) and (4.4.5). (In fact, this is what the SAGE command `zeta_polynomial` computes.)

SAGE

```
sage: C = HammingCode(3,GF(2))
sage: C.zeta_polynomial()
2/5*T^2 + 2/5*T + 1/5
sage: C = best_known_linear_code(6,3,GF(2))
sage: C.minimum_distance()
3
sage: C.zeta_polynomial()
2/5*T^2 + 2/5*T + 1/5
```

4.4.4 Analogies with Curves

Let X be a smooth projective curve of genus[7] g over a finite field $GF(q)$. Suppose that X is defined by a polynomial equation $F(x, y) = 0$, where F is a polynomial with coefficients in $GF(q)$. Let N_k denote the number of solutions in $GF(q^k)$ and create the generating function

$$G(t) = N_1 t + N_2 t^2/2 + N_3 t^3/3 + \cdots.$$

Define the zeta function of X by the formal power series

$$\zeta(t) = \zeta_X(t) = \exp\big(G(t)\big), \tag{4.4.6}$$

so $Z(0) = 1$. In particular, the logarithmic derivative of $\zeta(t)$ has integral coefficients. It is known that[8]

$$\zeta_X(t) = \frac{p(t)}{(1 - t)(1 - qt)}$$

[7] These terms will not be defined precisely here. Please see Tsafsman and Vladut [TV], Sect. 2.3.2, or Schmidt [Sc] for a rigorous treatment.

[8] This was first proved by Dwork using p-adic methods [Dw].

with $p = p_X$ a polynomial of degree $2g$, where g is the genus of X. This has a "functional equation" of the form

$$p(t) = q^g t^{2g} p\left(\frac{1}{qt}\right).$$

The logarithmic derivative of ζ_X is the generating function of the sequence of counting numbers $\{N_1, N_2, \dots\}$. The Riemann hypothesis for curves over finite fields states that the roots of P have absolute value $q^{-1/2}$. It is well known that the Riemann hypothesis holds for ζ_X (so the roots of zeta function of a curve all have absolute value $1/\sqrt{q}$; this is a theorem of André Weil from the 1940s). Therefore, by a suitable change-of-variable (replacing t by $t/\sqrt{q}$), we see that curves over finite fields give rise to a large class of example of polynomials having roots on the unit circle. The paper of Kedlaya discusses approaches to finding such polynomials whose coefficients satisfy some arithmetic conditions.

These roots can be interpreted in terms of the eigenvalues of a linear transformation[9] on a vector space. In fact, there is a unitary symplectic $2g \times 2g$ matrix $\Theta = \Theta_X$ such that[10]

$$p(t) = \det\bigl(I - tq^{1/2}\Theta\bigr).$$

When C is a formally self-dual AG code (associated to a smooth projective curve X of genus g over a finite field, a divisor D on X and points $\{P_i\}$ on X disjoint from D; see, for example, [TV, TVN]) of genus g (as a code), the Duursma polynomial $P = P_C$ "often" has the same degree as $p = p_X$ and satisfies the same functional equation. One can see using Theorem 4.1.28 in [TVN] that such codes are rather easy to construct, so this situation is not too unusual. This motivates the following question.

Open Problem 19 Let C be a formally self-dual code over $GF(q)$. When is there a curve $X/GF(q)$ for which the zeta function of the curve ζ_X is equal (up to a constant factor, if necessary) to the zeta function Z_C of the code?

Since the Riemann hypothesis holds for ζ_X, a necessary condition for Open Question 19 to hold is that the Duursma zeta function of the code must satisfy the Riemann hypothesis. Generally, the Duursma zeta function of a self-dual code does not satisfy the Riemann hypothesis, but see Example 9.7 in [D6] for two (self-dual) codes for which this holds. Here is a SAGE computation which verifies this:[11]

[9] In fact, it is possible to interpret $P(t)$ in terms of the characteristic function of "the Frobenius operator" acting on a cohomology space, though we shall omit details here.

[10] See Faifman and Rudnick [FR] for an interesting analysis of the "statistics" of the eigenvalues of Θ in the case where X is "hyperelliptic."

[11] The reciprocal of the numerator of the ζ-function of a curve is the characteristic polynomial of the Frobenius endomorphism of the Jacobian, i.e., the Frobenius polynomial.

SAGE

```
sage: K = GF(2)
sage: E = EllipticCurve(K,[0,1,1,-2,0]); E
Elliptic Curve defined by y^2 + y = x^3 + x^2 over Finite Field
 of size 2
sage: E.trace_of_frobenius()
-2
sage: E.frobenius_polynomial()
x^2 + 2*x + 2
```

Remark 10 However, there are other reasons to question that these zeta functions agree except in unusual circumstances. For example, using Sect. 3.1.1 (especially, Corollary 3.1.13) in [TVN], one sees that $p(1)/p(0) = q/h$, where h is the so-called class number of X (which is the number of $GF(q)$-rational points on the Jacobian of X, [TVN], p. 135). On the other hand, $P(0)/P(1)$ is given in Corollary 97 above. It is possible that $q/h = (q-1)^{-1}\binom{n}{d}^{-1}A_d$, but, if true, this is highly nonintuitive.

Alain Connes and others have worked on a natural spectral interpretation of the zeros of the Riemann zeta function. In other words, one wants to construct a self-adjoint operator on a Hilbert space whose spectrum is the set of nontrivial zeros of the Riemann zeta function. In the analogy between the Riemann zeta function and the Hasse–Weil zeta function of a curve X, the analog of this self-adjoint operator is the Frobenius operator on a certain cohomology space. The next open question asks is there an analog for Duursma zeta functions as well?

Open Problem 20 Let C be a self-dual code over $GF(q)$. When is there a linear operator Φ on a "natural" rational vector space for which the zeta polynomial $P = P_C$ can be interpreted in terms of the characteristic function of Φ?

The coefficients of the logarithmic derivative of the Hasse–Weil zeta function of a curve $X/GF(q)$ are integers—they count the number of points of X over a certain extension field of $GF(q)$. Is there an analog for the Duursma zeta function?

Open Problem 21 Let C be a self-dual code over $GF(q)$. Is there a "natural" interpretation of the coefficients of the logarithmic derivative of Z_C? Does the logarithmic derivative of $Z_C(T)$ have integral coefficients?

There is a "natural" interpretation of the coefficients of P_C—see the construction in Sect. 4.4.3 above.

4.5 Properties

We survey some of the most remarkable properties, both conjectured and proven, of these zeta functions.

4.5.1 The Functional Equation

If $\gamma = \gamma(C)$ is the genus of C and if

$$z_C(T) = Z_C(T)T^{1-\gamma},$$

then the functional equation in [D1] can be written in the form

$$z_{C^\perp}(T) = z_C(1/qT).$$

If we let

$$\zeta_C(s) = Z_C\left(q^{-s}\right)$$

and

$$\xi_C(s) = z_C\left(q^{-s}\right),$$

then ζ_C and ξ_C have the same zeros, but ξ_C is "more symmetric" since the functional equation expressed in terms of it becomes[12]

$$\xi_{C^\perp}(s) = \xi_C(1-s).$$

Abusing terminology, we call both Z_C and ζ_C the *Duursma zeta function* of C.

The analog of this for a virtually self-dual weight enumerator is as follows: let F denote a virtually self-dual weight enumerator with degree n and minimum distance d, so $\gamma = n + 1 - k - d = n/2 + 1 - d$ is the genus.

In fact, since Duursma's zeta function *only* depends on C via its weight enumerator $A_C(x, y)$ of C, for any virtual weight enumerator $F(x, y)$, there is an associated *zeta function* $Z = Z_F$ and *zeta polynomial* $P = P_F$. If we define $F^\perp$ by $F^\perp = F \circ \sigma$, where

$$\sigma = \frac{1}{\sqrt{q}} \begin{pmatrix} 1 & q-1 \\ 1 & -1 \end{pmatrix},$$

then there is a functional equation relating Z and $Z^\perp = Z_{F^\perp}$ (and hence also P and $P^\perp = P_{F^\perp}$). Note that even though F may not depend on q, $F^\perp$ (and hence $Z^\perp$) does.

Proposition 111 *For any virtual weight enumerator F satisfying*

$$F(x, y) = a_0 M_{n,d}(x, y) + a_1 M_{n,d+1}(x, y) + \cdots + a_r M_{n,d+r}(x, y)$$

and for any q, the zeta function $Z = Z_F$ satisfies the functional equation

$$Z^\perp(T)T^{1-g^\perp} = Z\left(\frac{1}{qT}\right)\left(\frac{1}{qT}\right)^{1-g}. \tag{4.5.1}$$

[12]This notation is inspired by analogous notation used for functions associated with the classical Riemann zeta function. See any book on the Riemann zeta function or http://en.wikipedia.org/wiki/Riemann_zeta_function.

Analogously, the zeta polynomial $P = P_F$ satisfies the functional equation

$$P^{\perp}(T) = P\left(\frac{1}{qT}\right) q^{g} T^{g+g^{\perp}}, \tag{4.5.2}$$

where $g = n/2 + 1 - d$ and $g^{\perp} = n/2 + 1 - d^{\perp}$.

Remark 11 (1) Note that both $P^{\perp}$ and P are polynomials of degree $n + 2 - d - d^{\perp} = g + g^{\perp}$ and g is the genus if $F = A_C$ is an actual weight enumerator.

(2) This proof is essentially the same as that of Proposition 9.2 in [D6]. This hypothesis here is slightly more general.

Proof This is a consequence of Definition 104 and the MacWilliams identity.

By hypothesis, the coefficients $a_j = a_j(F)$ of (4.4.3) satisfy $a_0 + \cdots + a_r = 1$. Therefore, $F^{\perp} = F \circ \sigma$ satisfies

$$F^{\perp} = a_0 M_{n,d} \circ \sigma + a_1 M_{n,d+1} \circ \sigma + \cdots + a_r M_{n,d+r} \circ \sigma. \tag{4.5.3}$$

Recall that the dual of the MDS code with parameters $[n, k, \delta]$ is the MDS code with parameters $[n, k^{\perp}, \delta^{\perp}]$. By this and MacWilliams' identity, we have $M_{n,\delta} \circ \sigma = q^{n/2+1-\delta} M_{n,\delta^{\perp}} = q^{k-n/2} M_{n,\delta^{\perp}}$, where $k^{\perp} + \delta^{\perp} = n + 1$, and $k = n - \delta + 1$ is the dimension of the (virtual) MDS code of length n and minimum distance δ (for a proof of this, see Appendix A in Duursma [D5]). Thus, $M_{n,\delta} \circ \sigma = q^{n/2+1-\delta} M_{n,n-\delta+2}$, and it follows that

$$\begin{aligned} F^{\perp} &= \sum_{d \le \delta \le d+r} a_{\delta-d} q^{n/2+1-\delta} M_{n,n-\delta+2} \\ &= \sum_{n-d-r+2 \le \delta' \le n-d+2} a_{n-\delta'+2-d} q^{\delta'-1-n/2} M_{n,\delta'} \\ &= \sum_{0 \le \delta'' \le r} a_{r-\delta''} q^{n/2-d-r+1+\delta''} M_{n,n-d-r+2+\delta''}. \end{aligned}$$

This implies

$$\begin{aligned} P^{\perp}(T) &= a_0^{\perp} + a_1^{\perp} T + \cdots + a_r^{\perp} T^r \\ &= a_r q^{n/2-r-d+1} + a_{r-1} q^{n/2-r-d+2} T + \cdots + a_0 q^{n/2-d+1} T^r \\ &= a_r q^{n/2-r-d+1} + a_{r-1} q^{n/2-r-d+1} (Tq) + \cdots + a_0 q^{n/2-r-d+1} (Tq)^r \\ &= q^{n/2-r-d+1} \left(a_r + a_{r-1}(Tq) + \cdots + a_0 (Tq)^r\right) \\ &= q^{n/2-r-d+1} (Tq)^r \left(a_0 + a_1 (Tq)^{-1} + \cdots + a_r (Tq)^{-r}\right) \\ &= q^{n/2-d+1} T^r P(1/qT). \end{aligned}$$

□

4.5.2 Puncturing Preserves P

Suppose that C is an $[n, k, d]$ code over $GF(q)$ and i is any integer satisfying $1 \leq i \leq n$. The *punctured code* $P_i(C)$ at the coordinate i is the code having length $n-1$ obtained by projecting C onto the remaining coordinates. The *shortened code* $S_i(C)$ at the coordinate i is the code having length $n-1$ obtained by projecting the subcode

$$\{c = (c_1, \ldots, c_n) \in C \mid c_i = 0\}$$

onto the remaining coordinates.

Lemma 112 *If C is a linear code of length n and i is an integer, $1 \leq i \leq n$, then*

$$P_i(C)^{\perp} = S_i(C^{\perp}).$$

A *check-bit extension* $\hat{C}$ is a code of length $n+1$ of the form

$$\{(c_1, \ldots, c_n, c_{n+1}) \in GF(q)^{n+1} \mid (c_1, \ldots, c_n) \in C,\ c_{n+1} = c \cdot a\}$$

for some fixed vector $a \in GF(q)^n$.

To end this section, we recall that the zeta polynomial of a code C, P_C, remains the same if we replace C by (a) the averaged puncturing $P(C)$ of C, (b) the averaged shortening $S(C)$ of C, or (c) a check-bit extension $\hat{C}$ of C. These facts provide inductive formulas for computing the zeta polynomial.

Theorem 113 (Duursma [D5]) *If C is a linear code of length n, if*

$$F_{P(C)}(x, y) = \frac{1}{n} \sum_{i=1}^{n} A_{P_i(C)}(x, y)$$

denotes the averaged punctured weight enumerator, and if

$$F_{S(C)}(x, y) = \frac{1}{n} \sum_{i=1}^{n} A_{S_i(C)}(x, y)$$

denotes the averaged shortened weight enumerator, then

$$P_C(T) = P_{F_{P(C)}}(T) = P_{F_{S(C)}}(T).$$

This is proven in Sect. 5 of Duursma [D5].

4.5.3 The Riemann Hypothesis

Knowledge of the zeros of $Z(T)$ could be very useful for understanding the possible values of the minimum distance. Let C be a code which is not MDS. If $\rho_1, \rho_2, \ldots, \rho_r$

denote the zeros, counted according to multiplicity, of the Duursma zeta polynomial $P(T)$ of a linear code C, then

$$\frac{P'(T)}{P(T)} = \sum_i \frac{1}{T - \rho_i}.$$

Proposition 114 (Duursma) *If $[A_0, \ldots, A_n]$ denotes the spectrum of C. then*

$$d = q - \sum_i \rho_i^{-1} - \frac{A_{d+1}}{A_d} \frac{d+1}{n-d}.$$

In particular,

$$d \leq q - \sum_i \rho_i^{-1}.$$

The proof uses the assumption that C is an actual linear code, not a virtual code, and that $P(T) \neq 1$.

Proof For the first statement, see Corollary 97. The second statement follows from the first since $\frac{A_{d+1}}{A_d} \geq 0$. □

In particular, if C is any b-divisible code with $b \geq 2$, then

$$d = q - \sum_i \rho_i^{-1}. \tag{4.5.4}$$

If F is a virtually self-dual weight enumerator, then the zeros of the zeta function $\zeta_F(s)$ (or $\xi_F(s)$) occur in pairs about the "critical line" $\mathrm{Re}\,(s) = \frac{1}{2}$.

Definition 115 We say the zeta function ζ_F (or, by abuse of terminology, the virtually self-dual weight enumerator F) satisfies the *Riemann hypothesis* (if all zeta zeros occur on the "critical line."

The following result is not best possible, but illustrates the idea that for "large" q, the Riemann hypothesis is "often" false.

Corollary 116 *Let C be an $[n, k, d]$ code over $GF(q)$ with $A_{d+1} = 0$, $q > n^2$, $2 \leq d$, and $d + d^\perp < n + 2$. If $n > 3$, then the Duursma zeta polynomial is not a constant and does not satisfy the Riemann hypothesis.*

This is an easy consequence[13] of Proposition 114, and the proof is left to the reader. The hypotheses to this corollary are probably not best possible. The point is that it should not be hard to construct codes which violate the Riemann hypothesis.

[13] Assume that the Riemann hypothesis is true and $q > n^2$. Then show that the hypothesis contradicts the trivial estimate $q - d \leq |\sum_i \rho_i^{-1}| \leq r\sqrt{q} = (n + 2 - d - d^\perp)\sqrt{q}$.

Example 117 It is clear from Example 105 above that the Duursma zeta function may have no zeros (i.e., may be constant). Indeed, this is true for all MDS codes, including some formally self-dual ones.[14]

Remark 12 Let F denote a virtually self-dual weight enumerator as in Proposition 111, and let $r(T) = z_F(T/\sqrt{q}\,)$. The functional equation implies that $r(T)$ is a self-reciprocal function: $r(1/T) = r(T)$. The Riemann hypothesis is the statement that all 2γ zeros of $r(T)$ lie on the "critical line" $|T| = 1$. If $r_0(\theta) = r(e^{i\theta})$, then the functional equation and the fact that r has rational coefficients imply

$$r_0(\theta) = r_0(-\theta) = \overline{r_0(\theta)}.$$

In other words, $r_0(\theta)$ is real valued.

The following open question is all the more tantalizing because we actually know (thanks to Duursma [D3]) explicitly the Duursma zeta functions of all extremal virtually self-dual weight enumerators.

Open Problem 22 (Duursma) For all extremal virtual (self-dual) weight enumerators F, the zeta function $Z = Z_F$ satisfies the Riemann hypothesis.

This is the *Riemann hypothesis* for virtually self-dual weight enumerators.

Lemma 118 *Let F denote a virtually self-dual weight enumerator of genus γ as above, and let $P = P_F$ denote the associated zeta polynomial. It is known that $P(T^2/q) = T^{2\gamma} f(T + T^{-1})$, where $f \in \mathbb{R}[x]$ is a polynomial of degree 2γ with real coefficients.*

Proof See Duursma [D3], Theorem 7 and Lemma 10. □

4.6 Self-reciprocal Polynomials

Thanks to the functional equation for the Duursma zeta polynomial, the validity of the Riemann hypothesis for a self-dual code (more generally a virtual self-dual weight enumerator) can be reduced to the question of whether or not a related polynomial has all its zeros on the unit circle. This section contains some of the basic results known about zeros of self-reciprocal polynomials on the unit circle.

[14] Formally self-dual MDS codes exist—see Example 12 in [JKT], which gives a formally self-dual $[42, 21, 22]$-code over a very large extension of $GF(7)$. (In fact, this code even has A_5 as its permutation automorphism group.) Even better, in Kim and Lee [KL], a self-dual MDS code with parameters $[10, 5, 6]_{41}$ is constructed.

4.6.1 "Smoothness" of Roots

A natural question to ask about zeros of polynomials is how "smoothly" do they vary as functions of the coefficients of the polynomial?

To address this, suppose that the coefficients a_i of the polynomial p are functions of a real parameter t. Abusing notation slightly, identify $p(z) = p(t, z)$ with a function of two variables ($t \in \mathbb{R}$, $z \in \mathbb{C}$). Let $r = r(t)$ denote a root of this polynomial, regarded as a function of t:

$$p\big(t, r(t)\big) = 0.$$

Using the two-dimensional chain rule,

$$0 = \frac{d}{dt} p\big(t, r(t)\big) = p_t\big(t, r(t)\big) + r'(t) \cdot p_z\big(t, r(t)\big),$$

so $r'(t) = -p_t(t, r(t))/p_z(t, r(t))$. Since $p_z(t, r(t)) = p'(r)$, the denominator of this expression for $r'(t)$ is zero if and only if r is a double root of p (i.e., a root of multiplicity 2 or more).

In answer to the above question, we have proven the following result on the "smoothness of roots."

Lemma 119 *$r = r(t)$ is smooth (i.e., continuously differentiable) as a function of t, provided that t is restricted to an interval on which $p(t, z)$ has no double roots.*

Consider the distance function

$$d(t) = \big|r(t)\big|$$

of the root r. Another natural question is: How smooth is the distance function of a root as a function of the coefficients of the polynomial p?

The analog to Lemma 119 holds with one extra condition.

Lemma 120 *$d(t) = |r(t)|$ is smooth (i.e., continuously differentiable) as a function of t, provided that t is restricted to an interval one which $p(t, z)$ has no double roots and $r(t) \neq 0$.*

Proof This is basically an immediate consequence of the above lemma and the chain rule,

$$\frac{d}{dt}\big|r(t)\big| = r'(t) \cdot \left(\frac{d|x|}{dx}\bigg|_{x=r(t)} \right). \qquad \square$$

4.6.2 Variations on a Theorem of Eneström–Kakeya

The following theorem was discovered independently by Eneström (in the late 1800s) and Kakeya (in the early 1900s).

Theorem 121 (Eneström–Kakeya, Version 1) *Let* $f(T) = a_0 + a_1T + \cdots + a_kT^k$ *satisfy* $a_0 > a_1 > \cdots > a_k > 0$. *Then* $f(T)$ *has no roots in* $|T| \leq 1$.

Remark 13 Replacing the polynomial by its reverse, here is "version 2" of the Eneström–Kakeya theorem: Let $f(z) = a_0 + a_1z + \cdots + a_kz^k$ satisfy $0 < a_0 < a_1 < \cdots < a_k$. Then $f(z)$ has no roots in $|z| \geq 1$.

An interesting discussion on the "sharpness" of this result (i.e., to what extent a converse theorem holds) can be found in Anderson, Saff, and Varga [ASV].

Below, we state Chinen's lemma, discovered independently by W. Chen,[15] whose proof is sketched in the next section (see also [Ch3]).

Corollary 122 (Chen–Chinen) *If* $f(T)$ *is a degree* m *polynomial of "decreasing symmetric form"*

$$f(T) = a_0 + a_1T + \cdots + a_kT^k + a_kT^{m-k} + a_{k-1}T^{m-k+1} + \cdots + a_0T^m$$

with $a_0 > a_1 > \cdots > a_k > 0$, *then all roots of* $f(T)$ *lie on the unit circle* $|T| = 1$, *provided that* $m \geq k$.

4.6.3 A Literature Survey

We recall some facts about self-reciprocal polynomials having roots on the unit circle from papers of Ancochea [An], Anderson, Saf, and Varga [ASV] Bonsall and Marden [BoM], Chen [Chen], Chinen [Ch3], DiPippo and Howe [DH], Fell [Fe], Kedlaya [Ked], S.-L. Kim [K], Kim and Park [KiP], Konvalina and Matache [KM], works of Lakatos and Losonczi [L1, L2, LL1, LL2], Petersen and Sinclair [PS], and Schiznel [Scl].

There is also a closely related body of research on Littlewood polynomials (which may have in fact motivated many of the papers listed above), for example, Drungilas [Dr] or Mercer [M]. These papers are related to the investigation of the "Littlewood problem" in connection with autocorrelation of binary sequences. However, the Littlewood polynomials are sufficiently different from the (suitably normalized) Duursma zeta polynomials that we shall have no need to refer further to those results.

For example, Lemma 2.1.1 in DiPippo and Howe [DH] provides one way of classifying those polynomials of even degree in $\mathbb{R}[x]$ which have all its roots on the unit circle. That result is discussed below following some preliminary definitions.

A polynomial p of the form

$$p(z) = \sum_{j=0}^{m} a_j z^j,$$

[15] Actually, Chen found a somewhat stronger result—see Theorem 134 below for a special case.

where $m \geq 1$, $a_m \neq 0$, $a_0, \ldots, a_m \in \mathbb{C}$, and $a_j = a_{m-j}$ $(0 \leq j \leq m/2)$, is called a *self-reciprocal polynomial* of degree m. Define the *reciprocal* or *reverse* polynomial of p by

$$p^*(z) = z^{\deg(p)} \cdot p(1/z), \tag{4.6.1}$$

where p is a polynomial of degree $\deg(p)$. Denote by $\mathbb{R}[z]_m$ the polynomials of degree $\leq m$ with real coefficients.

$$\mathbb{R}[z]_m = \big\{ p \in \mathbb{R}[z] \mid \deg(p) \leq m \big\}. \tag{4.6.2}$$

Denote by R_m the self-reciprocal polynomials of degree $\leq m$ with real coefficients,

$$R_m = \big\{ p \in \mathbb{R}[z]_m \mid p = p^* \big\}.$$

If p is a self-reciprocal polynomial of degree $2n$, then

$$p(z) = \sum_{j=0}^{2n} a_j z^j = z^n \big[a_{2n}\big(z^n + z^{-n}\big) + \cdots + a_{n+1}\big(z + z^{-1}\big) + a_n \big].$$

This shows that if β is a zero of p, then so is $1/\beta$.

The following statement is proven in Lakatos [L2]:

Lemma 123 *For each $p \in R_{2n}$ of degree $2n$ with $a_{2n} \neq 0$, there are n real numbers $\alpha_1, \ldots, \alpha_n$ such that*

$$p(z) = a_{2n} \prod_{k=0}^{n} \big(z^2 - \alpha_k z + 1\big). \tag{4.6.3}$$

The *Chebyshev transformation* $T : R_{2n} \to \mathbb{R}[z]_n$ is defined on the subset[16] of polynomials of degree $2n$ by

$$T_p(x) = a_{2n} \prod_{k=0}^{n} (x - \alpha_k),$$

where $x = z + z^{-1}$, and p and α_i are as in (4.6.3).

The following statement is proven in Lakatos [L2].

Lemma 124 *The Chebyshev transformation $T : R_{2n} \to \mathbb{R}[z]_n$ is a vector space isomorphism.*

[16] For simplicity, in this definition, we assume that $a_{2n} \neq 0$; see [L2] for the general definition of T.

For any $X_i \in \mathbb{C}$ $(1 \le i \le n)$, let

$$\begin{aligned}
e_0(X_1, X_2, \ldots, X_n) &= 1,\\
e_1(X_1, X_2, \ldots, X_n) &= \sum_{1\le j\le n} X_j,\\
e_2(X_1, X_2, \ldots, X_n) &= \sum_{1\le j<k\le n} X_j X_k,\\
e_3(X_1, X_2, \ldots, X_n) &= \sum_{1\le j<k<l\le n} X_j X_k X_l,\\
&\vdots\\
e_n(X_1, X_2, \ldots, X_n) &= X_1 X_2 \cdots X_n.
\end{aligned}$$

The following result is proven in Losonczi [Los].

Lemma 125 *For all $n \ge 1$ and $\alpha_i \in \mathbb{C}$, we have*

$$\prod_{k=0}^{n} \left(z^2 - \alpha_k z + 1\right) = \sum_{k=1}^{2n} c_{2n,k} z^k,$$

where $c_{2n,k} = c_{2n,2n-k}$ and

$$c_{2n,k} = (-1)^k \sum_{\ell=1}^{[k/2]} \binom{n-k+2\ell}{\ell} e_{k-2\ell}(\alpha_1, \ldots, \alpha_n)$$

for $0 \le k \le n$.

The following statement is proven in DiPippo and Howe [DH] (and found independently by Losonczi [Los]).

Lemma 126 *A polynomial $p \in R_{2n}$ has all its zeros on the unit circle if and only if there are n real numbers $\alpha_1, \ldots, \alpha_n$ in the interval $[-2, 2]$ such that (4.6.3) holds.*

Remark 14 If $p(z) \in R_n$ is a self-reciprocal monomial polynomial having all its coefficients lying on the unit circle, then p is determined by its $n-1$ coefficients. The topology and volume of those coefficients, regarded as a subset of $\mathbb{R}^{n-1}$, were recently determined by Petersen and Sinclair [PS].

Here is a different characterization, discovered by A. Cohn, of self-reciprocal polynomials having all roots on the unit circle.

Theorem 127 (Schur–Cohn) *Let $p(z)$ be a self-reciprocal polynomial of degree n. Suppose that $p(z)$ has exactly r zeros on the unit circle* (*counted according to multiplicity*) *and exactly s critical points in the closed unit disc* (*counted according to multiplicity*). *Then $r = 2(s+1) - n$.*

According to Chen [Chen], the above result of Cohn, published in 1922, is closely related[17] to a result of Schur, published in 1918. The following beautiful result is an immediate consequence.

Corollary 128 *A self-reciprocal polynomial polynomial has all its zeros on the unit circle if and only if all the zeros of its derivative lie inside or on the unit circle.*

See also Bonsall and Marsden [BoM] and Ancochea [An] (where they reprove a result of Cohn closely related to the theorem above).

There are various results in these papers which are, roughly speaking, stated as follows: if $p(z) \in R_{2n}$ is "near" a nonzero constant multiple of $1 + z + \cdots + z^{2n}$, then p has all its zeros on the unit circle. Here is an example of such a statement from Lakatos [L1].

Theorem 129 (Lakatos) *The polynomial $p \in R_{2n}$ given by*

$$p(z) = \ell\left(z^{2n} + z^{2n-1} + \cdots + z + 1\right) + \sum_{k=1}^{n} a_k\left(z^{2n-k} + z^k\right)$$

has all its roots on the unit circle if the coefficients satisfy the following condition:

$$|\ell| \geq 2\sum_{k=1}^{n} |a_k|.$$

A similar result holds for the odd-degree case (see [LL2]).

A statement in a similar framework, also due to Lakatos, is the following.

Theorem 130 (Lakatos) *The polynomial $p \in R_m$ given by*

$$p(z) = \sum_{j-0}^{m} a_j z^j$$

has all its roots on the unit circle if the coefficients satisfy the following condition:

$$|a_m| \geq \sum_{j=0}^{m} |a_j - a_m|.$$

[17] In fact, both are exercises in Marden [Ma].

Remark 15 (1) This result was generalized by Schiznel in 2005 [Scl] (the term $|a_j - a_m|$ was replaced by a more general linear combination).

(2) Of course, if $p(z)$ is very near the polynomials $1 + z + \cdots + z^m$, then the differences $a_j - a_m$ are very small, and the hypothesis obviously holds. In particular, this implies that self-reciprocal polynomials that are very near the polynomials $1 + z + \cdots + z^m$ have all their zeros on the unit circle.

Example 131 For example, according to SAGE, $f(z) = 1 + z + z^3 + z^4$ and $f(z) = 1 + z + z^2 + z^4 + z^5 + z^6$ have *all* their roots on the unit circle, but $f(z) = 1 + z + 2z^2 + 2z^4 + z^5 + z^6$ has only *some* (2 of its 6) roots on the unit circle.

Here is a detailed simple example to try to give some intuitive insight into the unusual results in the previous two theorems.

Example 132 Consider the polynomial

$$f_t(z) = 1 + (1 + t) \cdot z + z^2,$$

where $t \in \mathbb{R}$ is a parameter. Let $R(t)$ denote the set of roots of f_t, so

$$R(t) = \left\{ \frac{-1 - t \pm \sqrt{(1+t)^2 - 4}}{2} \right\},$$

and let

$$r(t) = \max_{z \in R(t)} \{|z|\}$$

be the size of the largest root. We plot this function $r(t)$. The claim is that $r(t)$ is not "smooth."

Note that if $0 < t < 1$, we have

$$r(t) = \left| \frac{-1 - t \pm i\sqrt{4 - (1+t)^2}}{2} \right| = \left(\frac{(1+t)^2}{4} + \frac{4 - (1+t)^2}{4} \right)^{1/2} = 1.$$

The plot[18] of $r(t)$ in the range $-5 < t < 3$ is in Fig. 4.1. This plot suggests that $r(t)$ is not differentiable. Indeed, if $t > 1$, then $r(t) = \frac{-1-t+\sqrt{(1+t)^2-4}}{2}$, so

$$r'(t) = -\frac{1}{2} + \frac{1+t}{\sqrt{(1+t)^2 - 4}}.$$

Note that $\lim_{t \to 1+} r'(t) = \infty$.

[18]The plot was created using SAGE's list_plot command, though the axes labels were modified using GIMP for ease of reading.

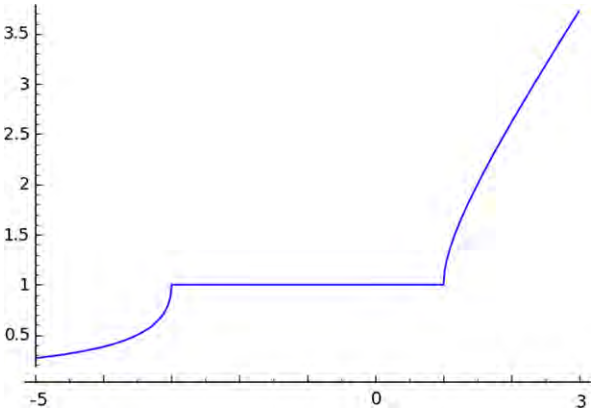


Fig. 4.1 Size of the largest root of the polynomial $1+(1+t)z+z^2$, $-5<t<3$

We will return to this topic in Sect. 4.6.1.

Corollary 133 *Consider a formally self-dual code C with associated zeta polynomial $P(T)=\sum_{i=0}^{2g}a_iT^i$ and "normalized" (self-reciprocal) zeta polynomial $R(T)=P(T/\sqrt{q})$. Write $R(T)=a_0\sum_{i=0}^{2g}c_iT^i$. If*

$$\sum_{k=1}^{n}|c_k-1|\leq\frac{1}{2},$$

then R has all its roots on the unit circle.

Remark 16 Note that $c_0=c_{2g}=1$ and that $c_i=a_iq^{-i/2}/a_0$ can in turn be related to the weights A_i via (4.4.1). Therefore, the above hypothesis implies a sort of growth condition on the coefficients a_i of P and hence also on the weights.

Recall that for a given polynomial $g(x)$ of degree d, $g^*(x)=x^dg(1/x)$ denotes the reciprocal polynomial. Note that if $f(x)=x^rg(x)+g^*(x)$, then $f^*(x)=f(x)$ ($r\geq 0$).

The following is basically the theorem of Chen and Chinen (Corollary 122).

Theorem 134 *If $0<a_0<\cdots<a_{k-1}<a_d$, then the roots of $x^rg(x)+g^*(x)$ all lie on the unit circle, $r\geq 0$.*

Proof We shall adapt some ideas from Chinen [Ch3] for our argument.

Write $f(T)$ as in (4.6.4) as

$$f(T)=g(T)+h(T),$$

where $g(T)=a_0+a_1T+\cdots+a_kT^k$ and $h(T)=a_kT^{m-k}+a_{k-1}T^{m-k+1}+\cdots+a_0T^m$. Given a polynomial $g(x)$, let $g^*(x)=x^kg(1/x)$ denote the reverse (or reciprocal) polynomial. Note that $h(T)=T^mg(T^{-1})=T^{m-k}g^*(T)$ and $f^*(T)=f(T)$.

Claim *$g^*(T)$ has no roots in $|T| \leq 1$.*

Proof This is equivalent to the statement of the Eneström–Kakeya theorem (Theorem 121). □

Claim *$g(T)$ has no roots in $|T| \geq 1$.*

Proof This follows from the previous claim and the observation that the roots of $g(T)$ correspond to the inverses of the roots of $g^*(T)$. □

Claim *$|g(T)| < |g^*(T)|$ on $|T| < 1$.*

Proof By the above claims, the function $\phi(T) = g(T)/g^*(T)$ is holomorphic on $|T| \leq 1$. Since $g(T^{-1}) = \overline{g(T)}$ on $|T| = 1$, we have $|g(T)| = |g^*(T)|$ on $|T| = 1$. The claim follows from the maximum modulus principle. □

Claim *The roots of $T^r g(T) + g^*(T)$ all lie on the unit circle, $r \geq 0$.*

Proof By the previous claim, $T^r g(T) + g^*(T)$ has the same number of zeros as $g^*(T)$ in the unit disc $|T| < 1$ (indeed, the function $\frac{T^r g(T)+g^*(T)}{g^*(T)} = 1 + \frac{T^r g(T)}{g^*(T)}$ has no zeros). Since $g^*(T)$ has no roots in $|T| < 1$, neither does $T^r g(T) + g^*(T)$. But since $T^r g(T) + g^*(T)$ is self-reciprocal (in this case), it has no zeros in $|T| > 1$ either. □

This proves Theorem 134. □

Here is a result which shows, in some sense, how close the Duursma zeta functions of extremal virtual codes are to polynomials which have *no* roots on the unit circle.

Lemma 135 *Let $f \in R_{2n}$, $f(x) = \sum_{i=0}^{d} c_i z^i$, d even, $c_0 < c_1 < \cdots < c_{d/2-1} < c_{d/2}$. If $2c_{d/2-1} < c_{d/2}$, then f has no roots on the unit circle. Conversely, if f has no roots on the unit circle, then $2c_0 < c_{d/2}$.*

For the converse, see Corollary 2 in Mercer [M]. For the proof of $\Longrightarrow$, we introduce the *Chebyshev polynomials* (of the first kind) T_k defined by

$$T_k(\cos\theta) = \cos(k\theta)$$

and their normalization $C_k(x) = 2T_k(x/2)$. It is known that

$$C_k\left(z + z^{-1}\right) = z^k + z^{-k}, \quad k > 0,$$

and we use the convention $C_0(x) = 1$.

Proof We can write

$$\frac{f(z)}{z^{d/2}} = z^{d/2}c_0\left(z^{d/2}+z^{-d/2}\right) + z^{d/2}\sum_{j=1}^{d/2-1} c_j\left(z^j+z^{d-j}\right)$$

$$= \sum_{j=0}^{d/2} c_j C_{d/2-j}\left(z+z^{-1}\right).$$

If $z = e^{i\theta}$, then

$$\sum_{j=0}^{d/2} c_j C_{d/2-j}(2\cos\theta) = c_{d/2} + 2\sum_{j=0}^{d/2} c_j \cos\left((d/2-j)\theta\right)$$

$$= \text{Real}\left[2\sum_{j=0}^{d/2} c_j \exp\left(i(d/2-j)\theta\right) - c_{d/2}\right]$$

$$= \text{Real}\left[2\sum_{j=0}^{d/2} c_j z^{d/2-j} - c_{d/2}\right].$$

If $2c_{d/2-1} < c_{d/2}$, then the Eneström–Kakeya theorem (Theorem 121) applies. □

If $P_0(z)$ and $P_1(z)$ are polynomials, let

$$P_a(z) = (1-a)P_0(z) + aP_1(z)$$

for $0 \le a \le 1$.

Next, we recall an interesting characterization due to Fell [Fe] (see also Kim [K] for discussion on a similar topic).

Theorem 136 (Fell) *Let $P_0(z)$ and $P_1(z)$ be real monic polynomials of degree n having zeros in $S^1 - \{1,-1\}$. Denote the zeros of $P_0(z)$ by $w_1, w_2, \ldots, w_n$ and of $P_1(z)$ by $z_1, z_2, \ldots, z_n$. Assume that*

$$w_i \neq z_j$$

for $1 \le i, j \le n$. Assume also that

$$0 < \arg(w_i) \le \arg(w_j) < 2\pi,$$

$$0 < \arg(z_i) \le \arg(z_j) < 2\pi,$$

for $1 \le i \le j \le n$. Let A_i be the smaller open arc of S^1 bounded by w_i and z_i for $1 \le i \le n$. Then the locus of $P_a(z)$, $0 \le a \le 1$, is contained in S^1 if and only if the arcs A_i are all disjoint.

4.6.4 Duursma's Conjecture

We say that a polynomial satisfying the condition

$$f(T) = a_0 + a_1 T + \cdots + a_k T^k + a_k T^{m-k} + a_{k-1} T^{m-k+1} + \cdots + a_0 T^m \quad (4.6.4)$$

with $a_k > a_{k-1} > \cdots > a_0 > 0$ has *increasing symmetric form.*[19]

If $m = 2k$ or $m = 2k+1$, then we say that $f(T)$ has *full support*.

There is an infinite family of Duursma zeta functions for which Duursma has conjectured that the analog of the Riemann hypothesis always holds. The linear codes used to construct these zeta functions are the so-called "extremal self-dual codes" (see Definition 92 for the more general notion of an extremal self-dual weight enumerator).

Although the construction of these codes is fairly technical (see [JK2] for an expository treatment), we can give some examples. They turn out to be of increasing symmetric form.

Example 137 Let $r(T) = \sum_i r_i T^i$ be as in Remark 12.

Some examples of the lists of coefficients $r_0, r_1, \ldots$ computed using SAGE. We have normalized the coefficients so that they sum to 10 and represented the rational coefficients as decimal approximations to give a feeling for their relative sizes.

- Case Type I:
  $m = 2$: [1.1309, 2.3990, 2.9403, 2.3990, 1.1309]
  $m = 3$: [0.45194, 1.2783, 2.0714, 2.3968, 2.0714, 1.2783, 0.45194]
  $m = 4$: [0.18262, 0.64565, 1.2866, 1.8489, 2.0724, 1.8489, 1.2866, 0.64565, 0.18262]
- Case Type II:
  $m = 2$: [0.43425, 0.92119, 1.3028, 1.5353, 1.6129, 1.5353, 1.3028, 0.92119, 0.43425]
  $m = 3$: [0.12659, 0.35805, 0.63295, 0.89512, 1.1052, 1.2394, 1.2854, 1.2394, 1.1052, 0.89512, 0.63295, 0.35805, 0.12659]
  $m = 4$: [0.037621, 0.13301, 0.28216, 0.46554, 0.65783, 0.83451, 0.97533, 1.0656, 1.0967, 1.0656, 0.97533, 0.83451, 0.65783, 0.46554, 0.28216, 0.13301, 0.037621]
- Case Type III:
  $m = 2$: [1.3397, 2.3205, 2.6795, 2.3205, 1.3397]
  $m = 3$: [0.58834, 1.3587, 1.9611, 2.1836, 1.9611, 1.3587, 0.58834]
  $m = 4$: [0.26170, 0.75545, 1.3085, 1.7307, 1.8874, 1.7307, 1.3085, 0.75545, 0.26170]

[19] The analogous definition of a polynomial of *decreasing symmetric form* also holds. The statement is left to the reader. The Eneström–Kakeya theorem implies (see Chinen's Theorem 122) that a polynomial of decreasing symmetric form has all its zeros on the unit circle.

- Case Type IV:
  $m = 2$: [2.8571, 4.2857, 2.8571]
  $m = 3$: [1.6667, 3.3333, 3.3333, 1.6667]
  $m = 4$: [0.97902, 2.4476, 3.1469, 2.4476, 0.97902]

Some remarks on the data in Example 137.

- Case Type I, $\nu = 0$: We conjecture that the coefficients of the (self-reciprocal) polynomial $R(T) = \sum_i r_i T^i$, where

$$\sum_i \binom{4m}{m+i} r_i T^i = \left(1 + T/\sqrt{2}\right)^m \left(1 + \sqrt{2}T\right)^m = \left(1 + 3T/\sqrt{2} + T^2\right)^m,$$

  have increasing symmetric form and full support.
- Case Type II, $\nu = 0$: We conjecture that the (self-reciprocal) polynomial $R(T) = \sum_i r_i T^i$, where

$$\sum_i \binom{6m}{m+i} r_i T^i = \left(1 + 2T/\sqrt{2} + T^2\right)^m \left(1 + 3T/\sqrt{2} + T^2\right)^m,$$

  has increasing symmetric form and full support.
- Case Type III, $\nu = 0$: We conjecture that the (self-reciprocal) polynomial $R(T) = \sum_i r_i T^i$, where

$$\sum_i \binom{4m}{m+i} r_i T^i = \left(1 + 3T/\sqrt{3} + T^2\right)^m$$

  has increasing symmetric form and full support.
- Case Type VI, $\nu = 0$: We conjecture that the (self-reciprocal) polynomial $R(T) = \sum_i r_i T^i$, where

$$\sum_i \binom{3m}{m+i} r_i T^i = (1 + T)^m,$$

  has increasing symmetric form and full support. The right-hand side has this property by well-known properties of the binomial coefficients.

4.6.5 A Conjecture on Zeros of Cosine Transforms

Are there conditions under which self-reciprocal polynomials with "increasing symmetric form" have all their zeros on S^1?

We know that self-reciprocal polynomial with "decreasing symmetric form" have all their roots on S^1 (by the Chen–Chinen theorem above). Under what conditions is the analogous statement true for functions with "increasing symmetric form?" The remainder of this section considers this question following [Jo2].

Let d be an odd integer, and let $f(z) = f_0 + f_1 z + \cdots + f_{d-1} z^{d-1} \in R_{d-1}$ be a self-reciprocal polynomial with "increasing symmetric form"

$$0 < f_0 < f_1 < \cdots < f_{\frac{d-1}{2}}.$$

For each $c \geq f_{\frac{d-1}{2}}$, the polynomial

$$g(z) = c \cdot \left(1 + z + \cdots + z^{d-1}\right) - f(z) = g_0 + g_1 z + \cdots + g_{d-1} z^{d-1} \in R_{d-1}$$

is a self-reciprocal polynomial having nonnegative coefficients with "decreasing symmetric form." If $c > f_{\frac{d-1}{2}}$, the Chen–Chinen theorem (Theorem 134) implies that all the zeros of $g(z)$ are on S^1. Let

$$P_0(z) = g(z)/g_{d-1}, \qquad P_1(z) = f(z)/f_{d-1}, \qquad P_a(z) = (1-a)P_0(z) + aP_1(z),$$

for $0 \leq a \leq 1$. By the Chen–Chinen theorem, there is a $t_0 \in (0, 1)$ such that all zeros of $P_t(z)$ are on S^1 for $0 \leq t < t_0$. In fact, if

$$t = \frac{f_{\frac{d-1}{2}} - f_{d-1}}{f_{\frac{d-1}{2}}},$$

then $P_t(z)$ is a multiple of $1 + z + \cdots + z^{d-1}$.

Do any of the polynomials $P_t(z)$ have multiple roots ($0 < t < 1$)? Using the notation of Sect. 4.6.1, in the case $p(t, z) = P_t(z)$, we have

$$r'(t) = -p_t\big(t, r(t)\big)/p_z\big(t, r(t)\big) = \frac{P_1(r(t)) - P_0(r(t))}{P_t'(r(t))}.$$

If no $P_t(z)$ has a multiple root, then by the second "smoothness-of-roots lemma" (Lemma 120), all the roots of $f(z)$ are also on S^1.

Conjecture 138 *Let $s : \mathbb{Z}_{>0} \to \mathbb{R}_{>0}$ be a "slowly increasing" function.*

- *Odd-degree case. If $g(z) = a_0 + a_1 z + \cdots + a_d z^d$, where $a_i = s(i)$, then the roots of $p(z) = g(z) + z^{d+1} g^*(z)$ all lie on the unit circle.*
- *Even-degree case. The roots of*

$$p(z) = a_0 + a_1 z + \cdots + a_{d-1} z^{d-1} + a_d z^d + a_{d-1} z^{d+1} + \cdots + a_1 z^{2d-1} + a_0 z^{2d}$$

all lie on the unit circle.

Using SAGE, one can guess that "logarithmic growth" might be "sufficiently slow."

SAGE

```
sage: R.<T> = PolynomialRing(CC,"T")
sage: c = [ln(j+2+random()) for j in range(5)];
sage: p = add([c[j]*T^j for j in range(5)])
        +T^5*add([c[4-j]*T^j for j in range(5)]); p
```

```
0.867252631954867*T^9 + 1.29158950186183*T^8 + 1.40385316206528*T^7
 + 1.66723678619336*T^6  + 1.79685924871722*T^5 + 1.79685924871722*T^4
 + 1.66723678619336*T^3 + 1.40385316206528*T^2 + 1.29158950186183*T
 + 0.867252631954867
sage: [z[0].abs() for z in p.roots()]
[1.00000000000000, 1.00000000000000, 1.00000000000000, 1.00000000000000,
 1.00000000000000, 1.00000000000000, 1.00000000000000, 1.00000000000000,
 1.00000000000000]
sage: c = [ln(j+2+random()) for j in range(5)]; c[4] = c[4]/2;
sage: p = add([c[j]*T^j for j in range(5)])
         +T^4*add([c[4-j]*T^j for j in range(5)]); p
1.07222251112144*T^8 + 1.34425116365361*T^7 + 1.55233692750212*T^6
 + 1.64078305774305*T^5 + 1.87422392028965*T^4 + 1.64078305774305*T^3
 + 1.55233692750212*T^2 + 1.34425116365361*T + 1.07222251112144
sage: [z[0].abs() for z in p.roots()]
[1.00000000000000, 1.00000000000000, 1.00000000000000, 1.00000000000000,
 1.00000000000000, 1.00000000000000, 1.00000000000000, 1.00000000000000]
```

4.7 Examples

4.7.1 Komichi's Example

In [HT], the authors mention an example which occurred in the master thesis[20] of A. Komichi. It is claimed that the Duursma zeta function of the code $C = H_8 \oplus H_8 \oplus H_8$, where H_8 is the self-dual extended Hamming $[8, 4, 4]$-code, violates the Riemann hypothesis. We verify this using SAGE.

SAGE

```
sage: MS = MatrixSpace(GF(2), 12, 24)
sage: G = MS([\
....: [ 1,1,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0 ],\
....: [ 0,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0 ],\
....: [ 0,0,1,0,1,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0 ],\
....: [ 0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0 ],\
....: [ 0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,1,0,0,0,0,0,0,0,0 ],\
....: [ 0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0 ],\
....: [ 0,0,0,0,0,0,0,0,0,0,1,0,1,1,0,1,0,0,0,0,0,0,0,0 ],\
....: [ 0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,0 ],\
....: [ 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,1 ],\
....: [ 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0 ],\
....: [ 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,0,1,1 ],\
....: [ 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,0,0 ]\
....: ])
sage: C = LinearCode(G)
sage: Cd = C.dual_code(); C == Cd
True
sage: R = PolynomialRing(CC,"T")
sage: T = R.gen()
sage: C.zeta_polynomial()
512/253*T^18 + 512/253*T^17 + 256/253*T^16 - 148736/245157*T^14
- 66048/81719*T^13 - 185536/245157*T^12 - 49408/81719*T^11
- 43088/96577*T^10 - 1808/5681*T^9 - 21544/96577*T^8 - 12352/81719*T^7
- 23192/245157*T^6 - 4128/81719*T^5 - 4648/245157*T^4 + 2/253*T^2
+ 2/253*T + 1/253
```

[20] This appears to be unpublished, and I have not seen it myself.

```
sage: f = R(C.zeta_polynomial())
sage: print [z[0].abs() for z in f.roots()]
[0.963950810639179, 0.707106781186546, 0.707106781186548,
 0.707106781186546, 0.518698666447988, 0.707106781186548,
 0.707106781186542, 0.707106781186548, 0.707106781186550,
 0.707106781186551, 0.707106781186547, 0.707106781186546,
 0.707106781186548, 0.707106781186544, 0.707106781186548,
 0.707106781186549, 0.707106781186548, 0.707106781186549]
sage: P1 = list_plot([(z[0].real(),z[0].imag()) for z in f.roots()])
sage: t = var("t")
sage: pts = lambda t: [cos(t)/sqrt(2),sin(t)/sqrt(2)]
sage: P2 = parametric_plot(pts(t),0,2*pi,linestyle="--",rgbcolor=(1,0,0))
sage: show(P1+P2)
```

The plot computed in the last line is shown in Fig. 4.2.

4.7.2 The Extremal Case

In this section, we shall summarize some results of Duursma [D3] and Harada and Tagami [HT].

If F is an extremal virtually self-dual weight enumerator, then the zeta function $Z = Z_F$ can be explicitly computed. First, some notation. If F is a virtually self-dual weight enumerator of minimum distance d and $P = P_F$ is its zeta polynomial,

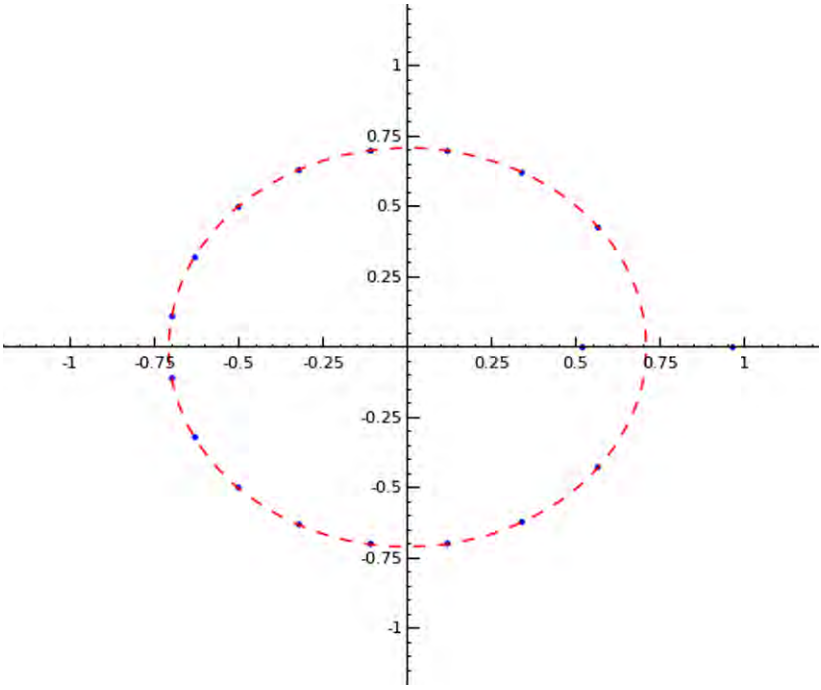


Fig. 4.2 Roots of the zeta polynomial for a self-dual [24, 12, 4] binary code

then define

$$Q(T) = \begin{cases} P(T), & \text{Type I,} \\ P(T)(1-2T+2T^2), & \text{Type II,} \\ P(T)(1+3T^2), & \text{Type III,} \\ P(T)(1+2T), & \text{Type IV.} \end{cases}$$

Let $(a)_m = a(a+1)\cdots(a+m-1)$ denote the *rising generalized factorial* and write $Q(T) = \sum_j q_j T^j$ for some $q_j \in \mathbb{Q}$. Let

$$\gamma_1(n,d,b) = (n-d)(d-b)_{b+1} A_d/(n-b-1)_{b+2}$$

and

$$\gamma_2(n,d,b,q) = (d-b)_{b+1} \frac{A_d}{(q-1)(n-b)_{b+1}},$$

where recall A_d denotes the coefficient of $x^{n-d}y^d$ in the virtual weight enumerator $F(x,y)$.

Theorem 139 (Duursma [D3]) *If F is an extremal virtually self-dual weight enumerator, then the coefficients of $Q(T)$ are determined as follows.*

(a) *If F is of Type I, then*

$$\sum_{i=0}^{2m+2\nu} \binom{4m+2\nu}{m+i} q_i T^i = \gamma_1(n,d,2)\cdot(1+T)^m(1+2T)^m\left(1+2T+2T^2\right)^\nu,$$

where $m = d-3$, $4m+2\nu = n-4$, $b = q = 2$, $0 \le \nu \le 3$.

(b) *If F is of Type II, then*

$$\sum_{i=0}^{4m+8\nu} \binom{6m+8\nu}{m+i} q_i T^i = \gamma_1(n,d,2)\cdot(1+T)^m(1+2T)^m\left(1+2T+2T^2\right)^m B(T)^\nu,$$

where $m = d-5$, $6m+8\nu = n-6$, $b = 4$, $q = 2$, $0 \le \nu \le 2$, and $B(T) = W_5(1+T,T)$, where W_5 is as in Example 44.

(c) *If F is of Type III, then*

$$\sum_{i=0}^{2m+4\nu} \binom{4m+4\nu}{m+i} q_i T^i = \gamma_2(n,d,3,3)\cdot\left(1+3T+3T^2\right)^m B(T)^\nu,$$

where $m = d-4$, $4m+4\nu = n-4$, $b = q = 3$, $0 \le \nu \le 2$, and $B(T) = W_9(1+T,T)$, where W_9 is as in Example 44.

(d) *If F is of Type IV, then*

$$\sum_{i=0}^{m+2\nu} \binom{3m+2\nu}{m+i} q_i T^i = \gamma_2(n,d,2,4)\cdot(1+2T)^m\big(1+2T+4T^2\big)^\nu,$$

where $m = d-3$, $3m+2\nu = n-3$, $b=2$, $q=4$, and $0 \le \nu \le 2$.

It is easy to determine (especially with a computer algebra system such as SAGE) the coefficients q_j and p_j from these expressions. So, for the virtual extremal codes, Duursma has computed all the Duursma zeta functions. Yet, we *still* do not know if the Riemann hypothesis holds for them!

Define the *ultraspherical polynomial* $C_n^m(x)$ on the interval $(-1,1)$ by

$$C_n^m(\cos\theta) = \sum_{\substack{0\le k,\ell\le n\\ k+\ell=n}} \binom{m+k}{k}\binom{m+\ell}{\ell}\cos(k-\ell)\theta.$$

Theorem 140 (Duursma [D3], Sect. 5.2)[21] *If P is the Duursma zeta polynomial of an extremal Type IV virtual self-dual weight enumerator of length $n = 3m+3$ and minimum distance $d = m+3$, then*

$$Q\big(T^2/2\big) = \frac{m!^2}{(3m)!}T^m C_m^{m+1}\left(\frac{T+T^{-1}}{2}\right).$$

(*Recall that, in this case, $Q(T) = P(T)(1+2T)$.*)

It is known that all the roots of ultraspherical polynomials C_n^m lie on the interval $(-1,1)$. The polynomial C_n^m is of degree n, and so there are n such roots. Replace T by $e^{i\theta}$ in the equation displayed in the theorem above to obtain

$$Q(e^{2i\theta}/2) = \frac{m!^2}{(3m)!}e^{i\theta m}C_m^{m+1}(\cos\theta).$$

Hence, all the roots of Q and therefore also of P lie on the circle of radius $1/\sqrt{q} = 1/2$. Indeed, the Riemann hypothesis holds for all zeta functions associated to an extremal Type IV virtually self-dual weight enumerator (Duursma [D3]).

Let $R(T) = P(T/\sqrt{q}) = \sum_{i=0}^{2g} r_i T^i$. This polynomial R is self-reciprocal. Though a lot is known about self-reciprocal polynomials which have all their zeros on the unit circle, we still do not know if the $P(T)$ satisfy the Riemann hypothesis or not! Duursma's approach is to try to describe the zeros of $H(z)$, where $R(T) = T^g H(T+T^{-1})$. By the theorem below, this function H can be explicitly described as a sum of ultraspherical polynomials. Though we know the zeros of the terms, we do not know the zeros of the sum in general. (The case of extremal codes of Type IV is different however.)

[21] A typo in [D3], Sect. 5.2, is corrected here.

Theorem 141 (Duursma [D3]) *If* α_j $(1 \le j \le g)$ *are defined by*

$$\sum_{i=0}^{2g} r_i \binom{2g+2d-4}{d-2+i} T^{2i} = T^{2g} \sum_{j=0}^{g} \alpha_j \binom{2j}{j}^{-1} (T+T^{-1})^{2j},$$

then

$$\binom{2g+2d-4}{g+d-2} \sum_{i=0}^{2g} r_i T^{2i} = T^{2g} \sum_{j=0}^{g} \alpha_j \binom{g+d-2}{j}^{-2} C_{2j}^{g+d-j-i} (T+T^{-1}).$$

Since $g+d = \frac{n}{2}+1$, these expressions can be simplified a bit, if desired. Also, in Sect. 5.2 in [D3], Duursma explicitly computes the α_j's in each case (Type I, II, III, and IV).

Using computer computations, Harada and Tagami [HT] (among other things) showed that the Riemann hypothesis holds for all zeta functions associated to extremal Type I, II, III virtually self-dual weight enumerators of degree ≤ 200.

4.7.3 "Random Divisible Codes"

Following Theorem 4 in Duursma [D5], we show that the Duursma zeta function of a "random divisible code" satisfies the Riemann hypothesis.

Define the *(virtual) weight enumerator of the* $[n,k]_q$ *random b-divisible code* by

$$F(x,y) = x^n + c \sum_{i=1}^{n/b} \binom{n}{ib} (q-1)^{bi} x^{n-bi} y^{bi},$$

where c is chosen so that $F(1,1) = q^k$, and n is a multiple of b. Of course, by the classification of b-divisible codes (see Theorem 89), this weight enumerator may not correspond to an actual linear code.

Duursma shows that in the following cases the zeta function $Z_F(T)$ satisfies the Riemann hypothesis: n is even, $k = n/2$, and

- $q = 2$, $b = 4$,
- $q = 3$, $b = 3$,
- $q = 4$, $b = 2$.

For details, see Duursma [D5], Theorem 4.

4.7.4 A Formally Self-dual $[26, 13, 6]_2$-code

Moreover, in this case the Riemann hypothesis is not valid for optimal codes (which may or may not be extremal) in general, as the following example illustrates.

Example 142 Consider the $[26, 13, 6]_2$ code with weight distribution

$$[1, 0, 0, 0, 0, 0, 39, 0, 455, 0, 1196, 0, 2405, 0, 2405, 0, 1196, 0, 455, 0, 39, 0, 0,$$
$$0, 0, 0, 1].$$

This is (by coding theory tables, as included in SAGE [S]) an optimal formally self-dual code. This code C has the zeta polynomial

$$\begin{aligned} P(T) = {} & \frac{3}{17710} + \frac{6}{8855}T + \frac{611}{336490}T^2 + \frac{9}{2185}T^3 + \frac{3441}{408595}T^4 + \frac{6448}{408595}T^5 \\ & + \frac{44499}{1634380}T^6 + \frac{22539}{520030}T^7 + \frac{66303}{1040060}T^8 + \frac{22539}{260015}T^9 + \frac{44499}{408595}T^{10} \\ & + \frac{51584}{408595}T^{11} + \frac{55056}{408595}T^{12} + \frac{288}{2185}T^{13} + \frac{19552}{168245}T^{14} + \frac{768}{8855}T^{15} \\ & + \frac{384}{8855}T^{16}. \end{aligned}$$

Using SAGE, it can be checked that only 8 of the 12 zeros of this function have absolute value $\sqrt{2}$.

4.7.5 Extremal Codes of Short Length

In this section, we give some examples using SAGE.

These do not satisfy $P(1) = 1$ but use the formulas in Theorem 139 above.

For the $[24, 12, 8]_2$ virtually self-dual weight enumerator:

$$\begin{aligned} P(T) = {} & \frac{2}{969}T^{10} + \frac{2}{323}T^9 + \frac{10}{969}T^8 + \frac{4}{323}T^7 + \frac{197}{16796}T^6 + \frac{9}{988}T^5 \\ & + \frac{197}{33592}T^4 + \frac{1}{323}T^3 + \frac{5}{3876}T^2 + \frac{1}{2584}T + \frac{1}{15504}. \end{aligned}$$

For the $[26, 13, 8]_2$ virtually self-dual weight enumerator:

$$\begin{aligned} P(T) = {} & \frac{32}{13167}T^{12} + \frac{32}{4389}T^{11} + \frac{4}{323}T^{10} + \frac{496}{31977}T^9 + \frac{393}{24871}T^8 + \frac{31}{2261}T^7 \\ & + \frac{281}{27132}T^6 + \frac{31}{4522}T^5 + \frac{393}{99484}T^4 + \frac{62}{31977}T^3 + \frac{1}{1292}T^2 + \frac{1}{4389}T \\ & + \frac{1}{26334}. \end{aligned}$$

For the $[28, 14, 8]_2$ virtually self-dual weight enumerator:

$$P(T) = \frac{16}{5313}T^{14} + \frac{16}{1771}T^{13} + \frac{224}{14421}T^{12} + \frac{96}{4807}T^{11} + \frac{3469}{163438}T^{10} + \frac{291}{14858}T^9 + \frac{23}{1428}T^8 + \frac{622}{52003}T^7 + \frac{23}{2856}T^6 + \frac{291}{59432}T^5 + \frac{3469}{1307504}T^4 + \frac{6}{4807}T^3 + \frac{7}{14421}T^2 + \frac{1}{7084}T + \frac{1}{42504}.$$

See also Example 137.

4.7.6 Non-self-dual Examples

Consider the optimal binary code C having the parameters $[6, 2, 4]$ and generator matrix

$$G = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

This has zeta polynomial $P(T) = (2T^2 + 2T + 1)/5$, as the following SAGE computation shows.

SAGE

```
sage: R_CC = PolynomialRing(CC,"T")
sage: n = 6; k = 2; q = 2
sage: C = best_known_linear_code(n,k,GF(q))
sage: C.zeta_polynomial()
2/5*T^2 + 2/5*T + 1/5
sage: [abs(z[0]) for z in R_CC(C.zeta_polynomial()).roots()]
[0.707106781186548, 0.707106781186548]
sage: C.weight_enumerator()
x^6 + 3*x^2*y^4
sage: Cd = C.dual_code()
sage: Cd.zeta_polynomial()
2/5*T^2 + 2/5*T + 1/5
sage: Cd.weight_enumerator()
x^6 + 3*x^4*y^2 + 8*x^3*y^3 + 3*x^2*y^4 + y^6
sage: n = 7; k = 4; q = 2
sage: C = best_known_linear_code(n,k,GF(q))
sage: C.zeta_polynomial()
2/5*T^2 + 2/5*T + 1/5
sage: C.weight_enumerator()
x^7 + 7*x^4*y^3 + 7*x^3*y^4 + y^7
sage: Cd = C.dual_code()
sage: Cd.zeta_polynomial()
2/5*T^2 + 2/5*T + 1/5
sage: Cd.weight_enumerator()
x^7 + 7*x^3*y^4
```

```
sage: n = 8; k = 4; q = 2
sage: C = best_known_linear_code(n,k,GF(q))
sage: C.zeta_polynomial()
2/5*T^2 + 2/5*T + 1/5
sage: C.weight_enumerator()
x^8 + 14*x^4*y^4 + y^8
sage: Cd = C.dual_code()
sage: Cd.zeta_polynomial()
2/5*T^2 + 2/5*T + 1/5
sage: Cd.weight_enumerator()
x^8 + 14*x^4*y^4 + y^8
```

Indeed, the optimal $[6, 2, 4]$ code has the same zeta polynomial as the Hamming $[7, 4, 3]$ code. This satisfies the Riemann hypothesis, although it is not formally self-dual. However, it does have the same zeta polynomial as the optimal self-dual $[8, 4, 4]$ code.

4.8 Chinen Zeta Functions

In the sections above, a virtual weight enumerator F is associated with a zeta function $Z = Z_F$. In this section, two related zeta functions were constructed by Koji Chinen. First, he constructed a zeta function $Z = Z_F$, which we call a "twisted Chinen zeta function," associated to a twisted virtually self-dual weight enumerator F. (What we call a "twisted virtually self-dual weight enumerator," he calls a "formal weight enumerator.") Next, he constructed a zeta function associated to any code C, which we call a "Chinen zeta function," which is essentially defined by combining the Duursma zeta function of C with that of its dual $C^\perp$ (some care is required to insure that the functional equation leads to an extra symmetry property).

Here is the analogous result for Chinen zeta functions of the results above.

Let C be any $[n, k, d]$ code over $GF(q)$, and let $[n, n-k, d^\perp]$ denote the parameters of the dual code $C^\perp$. We assume that they satisfy $d \geq 2$ and $d^\perp \geq 2$. Define the *invariant weight enumerator* by

$$\tilde{A}_C(x, y) = \frac{A_C(x, y) + q^{k-n/2} A_{C^\perp}(x, y)}{1 + q^{k-n/2}}.$$

Note that $\tilde{A}_C = \tilde{A}_{C^\perp} = \tilde{A}_C \circ \sigma_q$, by the MacWilliams identity. The *Chinen zeta polynomial* $\tilde{P}_C$ is the zeta polynomial P_F associated to the virtual weight enumerator $F = \tilde{A}_C$. The *Chinen zeta function* is defined in terms of the zeta polynomial by means of the following equation:

$$\tilde{P}_C(T) = \frac{T^{\max(0, d-d^\perp)}}{1 + q^{k-n/2}} \big(P_C(T) + q^{n/2-d+1} T^{n-2d+2} P_C(1/qT)\big). \qquad (4.8.1)$$

Theorem 143 (Chinen [Ch3]) *The Chinen zeta polynomial given by* (4.8.1) *above has degree* $2\tilde{g} = n + 2 - 2\min(d, d^\perp)$ *and satisfies the functional equation*

$$\tilde{P}_C(T) = q^{\tilde{g}} T^{2\tilde{g}} \tilde{P}_C(1/qT).$$

By the functional equation, if $d > d^\perp$, then

$$\tilde{P}_C(T) = \frac{q^{k-n/2} P_{C^\perp}(T) + T^{d-d^\perp} P_C(T)}{1 + q^{k-n/2}};$$

if $d < d^\perp$, then

$$\tilde{P}_C(T) = \frac{P_C(T) + q^{k-n/2} T^{d^\perp - d} P_{C^\perp}(T)}{1 + q^{k-n/2}};$$

and if $d = d^\perp$, then

$$\tilde{P}_C(T) = \frac{P_C(T) + q^{k-n/2} P_{C^\perp}(T)}{1 + q^{k-n/2}}.$$

Note that when $T = 1$, we have $P(1) = 1$ and (by the functional equation) $P(1/q) = q^{-g} = q^{d-1-n/2}$. This implies $\tilde{P}_C(1) = \frac{2}{1+q^{k-n/2}}$. It may be simpler to use the "averaged" zeta function

$$P_C^*(T) = \big(P_C(T) + P_{C^\perp}(T)\big)/2,$$

but this is *not* the Chinen zeta function.

Example 144 We use SAGE to compute the Chinen zeta polynomial of some small optimal codes. We shall normalize the Chinen zeta function so that $\tilde{P}_C(1) = 1$.

SAGE

```
sage: R_CC = PolynomialRing(CC,"T")
sage: n = 8; k = 2; q = 2
sage: C = best_known_linear_code(n,k,GF(q))
sage: P = C.chinen_polynomial()
sage: Cd = C.dual_code()
sage: Pd = Cd.chinen_polynomial()
sage: C.minimum_distance(); Cd.minimum_distance()
5
2
sage: P; P == Pd
2/5*t^6 + 9/35*t^5 + 4/35*t^4 + 2/35*t^3 + 2/35*t^2 + 9/140*t + 1/20
True
sage: [abs(z[0]) for z in R_CC(P*1.0).roots()]

[0.707106781186548,
 0.707106781186548,
 0.707106781186547,
 0.707106781186547,
 0.707106781186547,
 0.707106781186548]
sage: C.gen_mat()
```

```
[0 0 0 1 1 1 1]
[1 1 1 0 0 1 1 1]
sage: C0 = C.standard_form()[0]
sage: C0.gen_mat()
[1 0 1 1 0 1 1 1]
[0 1 0 0 1 1 1 1]
```

The Riemann hypothesis is (apparently) true since the zeros have absolute value (approximately) $1/\sqrt{2}$.

SAGE

```
sage: C = HammingCode(3,GF(2))
sage: C.chinen_polynomial()
(2*sqrt(2)*t^3/5 + 2*sqrt(2)*t^2/5 + 2*t^2/5
    + sqrt(2)*t/5 + 2*t/5 + 1/5)/(sqrt(2) + 1)
```

It can be easily shown that if C is formally self-dual, then $\tilde{P}_C = P_C$. We say that C (whether formally self-dual or not) *satisfies the Riemann hypothesis* if its Chinen zeta polynomial has all is zeros on the "critical line."

For example, if C is an MDS code, then

$$\tilde{P}_C(T) = \frac{1}{1+q^{k-n/2}}\left(1+q^{n/2-d+1}T^{n-2d+2}\right).$$

If C is MDS and $n-2d+2 \neq 0$, then the Riemann hypothesis holds for the Chinen zeta function.

The following is an analog of Open Question 19 for Chinen zeta functions.

Open Problem 23 Let C be any code over $GF(q)$. When is there a curve $X/GF(q)$ for which the zeta function of the curve ζ_X is equal to the Chinen zeta function Z_C of the code?

Since the Riemann hypothesis holds for ζ_X (this is a well-known theorem of André Weil), a necessary condition is that the code must satisfy the Riemann hypothesis. See Example 9.7 in [D6] for two (self-dual) codes for which this holds.

Remark 17 For the "twisted case," including detailed proofs and numerous examples, see Chinen [Ch2].

Open Problem 24 Is the Chinen zeta function of a linear code C equal to the Duursma zeta function of some self-dual code C'?

If yes, then of course the set of Chinen zeta functions would be contained in the set of Duursma zeta functions. For example, is the Chinen zeta function of a nonbinary Hamming code C (say over $GF(q)$ with $q > 4$) equal to the Duursma zeta function of some self-dual code C'? This seems unlikely, but we do not have a proof or disproof.

Example 145 We use SAGE to compute the Chinen zeta polynomial of some indecomposable codes.

Consider codes which are generated by the matrix D_m (m even) defined as follows.

SAGE

```
def d_matrix(m):
    if not(is_even(m)):
        raise ValueError, "%s must be even and >2"%m
    M = int(m/2)
    A = [[0]*2*i+[1]*4+[0]*(m-4-2*i) for i in range(M-1)]
    MS = MatrixSpace(GF(2), M-1, m)
    return MS(A)
```

For example,

$$D_{14} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix},$$

and the binary code generated by this matrix is a $[14, 6, 4]$ code. Using SAGE, you can see that the associated Chinen zeta function does not satisfy the Riemann hypothesis.

SAGE

```
sage: n = 14; G = d_matrix(n); C = LinearCode(G); C
Linear code of length 14, dimension 6 over Finite Field of size 2
sage: C.spectrum()
[1, 0, 0, 0, 21, 0, 0, 0, 35, 0, 0, 0, 7, 0, 0]
sage: PT = PolynomialRing(CC,"T")
sage: PC = C.chinen_polynomial(); rts = PT(PC).roots()
sage: PC
64/39*t^12 - 32/429*t^10 - 32/429*t^9 - 160/1287*t^8 - 64/429*t^7 -
160/1287*t^6 - 32/429*t^5 - 40/1287*t^4 - 4/429*t^3 - 2/429*t^2 + 1/39
sage: [z[0].abs() for z in rts]

[0.707106781186548,
 0.707106781186548,
 0.707106781186548,
 0.707106781186547,
 0.707106781186548,
 0.707106781186548,
 0.707106781186549,
 0.707106781186548,
 0.707106781186547,
 0.707106781186548,
 0.814795710093010,
 0.613650751723920]
```

In particular, the Riemann hypothesis for the Chinen zeta function is not true for all indecomposable codes.

4.8.1 Hamming Codes

Chinen [Ch3] computed the zeta polynomial of the Hamming codes. Consider the Hamming code $C = C_{r,q}$ having the parameters $[n = \frac{q^r-1}{q-1}, n-r, 3]$ over $GF(q)$, with $r \geq 3$. (When $r = 2$, the Hamming code is MDS and so has already been computed.)

The Duursma zeta polynomial of the dual code is given by

$$P_{C^\perp}(T) = c \cdot \left[1 + \sum_{j=1}^{n-d-1} \left(\binom{j+d-1}{d-1} - q \binom{j+d-2}{d-1} \right) T^j \right],$$

where the constant $c = c_{r,q}$ is chosen so that $P(1) = 1$. This is Proposition 4.4 in [Ch3].

The Chinen zeta polynomial of the Hamming codes $C_{r,q}$ ($r \geq 3$, $q \geq 2$) is given by

$$\tilde{P}_C(T) = \frac{c}{1+q^{r-n/2}} \big(F_1(T) - q F_2(T) \big), \tag{4.8.2}$$

where

$$F_1(T) = \sum_{j=0}^{n-d-1} \binom{n-i-2}{d-1} q^{i+2-n/2} T^i + \sum_{j=d-3}^{n-4} \binom{i+2}{d-1} T^i$$

and

$$F_2(T) = \sum_{j=0}^{n-d-2} \binom{n-i-3}{d-1} q^{i+2-n/2} T^i + \sum_{j=d-2}^{n-4} \binom{i+1}{d-1} T^i.$$

This is Theorem 4.5 in [Ch3].

Example 146 Here is the Chinen zeta polynomial of the Hamming $[7, 4, 3]$ code:

SAGE

```
sage: C = HammingCode(3,GF(2))
sage: C.chinin_polynomial()
(2*T^2/5 + 2*sqrt(2)*T*(T^2/5 + T/5 + 1/10) + 2*T/5 + 1/5)/(sqrt(2) + 1)
```

Theorem 147 (Chinen) *The Chinen zeta polynomial of the Hamming codes $C_{r,q}$ ($r \geq 3$, $q \geq 4$) satisfies the Riemann hypothesis.*

This theorem is also true when $r = 2$ ($q \geq 2$), as a corollary to (3.3) in [Ch3], since then C is MDS.

Chinen's proof of this theorem is beautiful and based on his result stated as Corollary 122 in Sect. 4.6.2 above. To prove Theorem 147, Chinen explicitly computes the coefficients a_i of a normalized Chinen zeta polynomial f of $C = C_{r,q}$ and proves

that it has the above decreasing symmetric form. This implies the Riemann hypothesis, as desired, The proof of the above lemma and the explicit computation of the coefficients are carefully worked out in [Ch3], which we refer to for details.

4.8.2 Golay Codes

This section summarizes some of the results in Chinen [Ch3], Sect. 7.

The Chinen zeta polynomial of the $[11, 6, 5]$ Golay code C over $GF(3)$ is

$$\tilde{P}_C(T) = \frac{\sqrt{3}-1}{14}(\sqrt{3}T+1)\left(3T^2+3T+1\right).$$

Chinen also presents an explicit expression but complicated expression for the Chinen zeta polynomial of the $[23, 12, 7]$ Golay code C over $GF(2)$. He also shows that both of these Chinen zeta functions satisfy the "Riemann hypothesis." The proof is by explicitly computing zeros, verifying the Riemann hypothesis numerically.

4.8.3 Examples

We begin with a random example:

SAGE

```
sage: RT = PolynomialRing(CC,"T")
sage: MS = MatrixSpace(GF(2), 3, 8)
sage: G = MS([[1,0,0,1,0,1,1,0],[0,1,0,1,0,0,0,1],[0,0,1,0,1,1,1,0]])
sage: C = LinearCode(G)
sage: C.minimum_distance()
3
sage: Cd = C.dual_code(); Cd.minimum_distance()
2
sage: f = RT(C.chinen_polynomial())
sage: print [z[0].abs() for z in f.roots()]
[0.707106781186548, 0.707106781186548, 0.707106781186548,
 0.707106781186548, 0.707106781186547, 0.707106781186547]
sage: C.gen_mat()
[1 0 0 1 0 1 1 0]
[0 1 0 1 0 0 0 1]
[0 0 1 0 1 1 1 0]
sage: C.spectrum()
[1, 0, 0, 1, 3, 2, 0, 1, 0]
sage: Cd.spectrum()
[1, 0, 3, 10, 7, 4, 5, 2, 0]
sage: C.chinen_polynomial()
2/7*t^6 + 4/21*t^5 + 13/70*t^4 + 17/105*t^3 + 13/140*t^2 + 1/21*t + 1/28
sage: C.zeta_polynomial()
3/7*T^5 + 3/14*T^4 + 11/70*T^3 + 17/140*T^2 + 17/280*T + 1/56
sage: f = RT(C.zeta_polynomial())
sage: print [z[0].abs() for z in f.roots()]
[0.644472635143760, 0.644472635143761, 0.458731710756610,
 0.476718789722295, 0.458731710756610]
```

This next example is also random:

SAGE

```
sage: C = RandomLinearCode(8,3,GF(2)); C.minimum_distance()
3
sage: Cd = C.dual_code(); Cd.minimum_distance()
2
sage: C.spectrum()
[1, 0, 0, 1, 3, 2, 0, 1, 0]
sage: Cd.spectrum()
[1, 0, 3, 6, 11, 8, 1, 2, 0]
sage: C.chinen_polynomial()
2/7*t^6 + 4/21*t^5 + 13/70*t^4 + 17/105*t^3 + 13/140*t^2 + 1/21*t + 1/28
sage: C.gen_mat()
[1 0 0 1 1 0 0 1]
[0 1 0 0 0 1 1 0]
[0 0 1 1 0 0 1 1]
sage: C.zeta_polynomial()
3/7*T^5 + 3/14*T^4 + 11/70*T^3 + 17/140*T^2 + 17/280*T + 1/56
```

The next example concerns a code which is formally self-dual but not self-dual.

SAGE

```
sage: RT = PolynomialRing(CC,"T")
sage: MS = MatrixSpace(GF(2), 4, 8)
sage: G = MS([[1,0,0,0,0,1,1,0],[0,1,0,0,1,1,1,0],
              [0,0,1,0,1,1,1,1],[0,0,0,1,0,0,1,0]])
sage: C = LinearCode(G)
sage: C.minimum_distance()
2
sage: Cd = C.dual_code(); Cd.minimum_distance()
2
sage: f = RT(C.chinen_polynomial())
sage: print [z[0].abs() for z in f.roots()]
[0.707106781186549, 0.707106781186547, 0.707106781186547,
 0.707106781186546, 0.707106781186547, 0.707106781186547]
sage: C.gen_mat()
[1 0 0 0 0 1 1 0]
[0 1 0 0 1 1 1 0]
[0 0 1 0 1 1 1 1]
[0 0 0 1 0 0 1 0]
sage: C.chinen_polynomial()
2/7*t^6 + 2/7*t^5 + 11/70*t^4 + 3/35*t^3 + 11/140*t^2 + 1/14*t + 1/28
sage: C.spectrum()
[1, 0, 1, 4, 3, 4, 3, 0, 0]
sage: Cd = C.dual_code(); Cd.minimum_distance()
2
sage: Cd.spectrum()
[1, 0, 1, 4, 3, 4, 3, 0, 0]
sage: list_plot([(z[0].real(),z[0].imag()) for z in f.roots()])
```

The last command gives a plot of the roots (see Fig. 4.3).

Our last example is one for which the Riemann hypothesis is false.

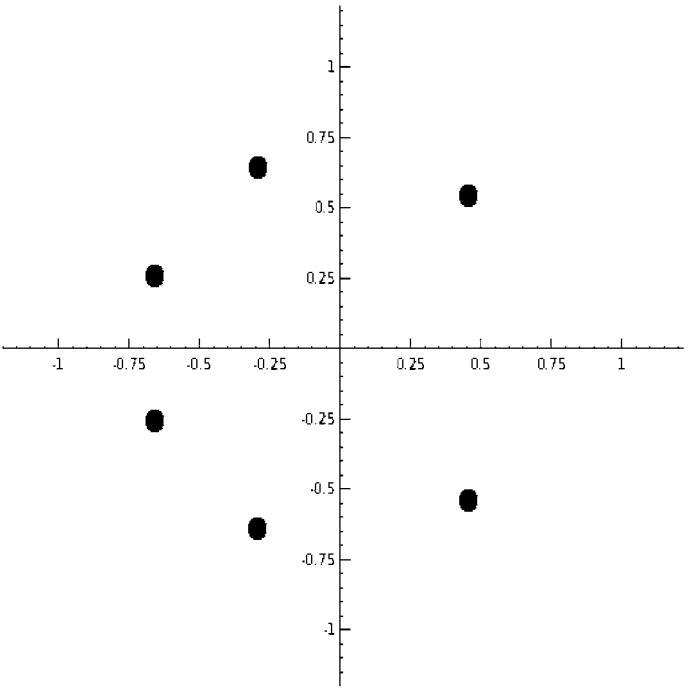


Fig. 4.3 Roots of the Chinen zeta polynomial for a formally self-dual $[8, 4, 2]$ binary code

SAGE

```
sage: RT = PolynomialRing(CC,"T")
sage: MS = MatrixSpace(GF(2), 4, 8)
sage: G = MS([[1,1,0,0,0,0,1,1],[0,0,1,0,0,1,0,1],[0,0,0,1,0,1,1,0],
             [0,0,0,0,1,1,1,1]])
sage: C = LinearCode(G)
sage: C.chinen_polynomial()
1/7*t^6 + 1/7*t^5 + 39/140*t^4 + 17/70*t^3 + 39/280*t^2 + 1/28*t + 1/56
sage: C.spectrum()
[1, 0, 0, 4, 6, 4, 0, 0, 1]
sage: Cd = C.dual_code(); Cd.minimum_distance()
2
sage: Cd.spectrum()
[1, 0, 1, 0, 11, 0, 3, 0, 0]
sage: C.minimum_distance()
3
sage: Cd = C.dual_code(); Cd.minimum_distance()
2
sage: f = RT(C.chinen_polynomial())
sage: print [z[0].abs() for z in f.roots()]
[1.19773471696883, 1.19773471696883, 0.707106781186547,
 0.707106781186547, 0.417454710894058, 0.417454710894058]
sage: print [z[0] for z in f.roots()]
[0.0528116723604142 + 1.19656983895421*I,
 0.0528116723604137 - 1.19656983895421*I,
 -0.571218487412783 + 0.416784644196318*I,
 -0.571218487412783 - 0.416784644196317*I,
 0.0184068150523700 + 0.417048707955401*I,
 0.0184068150523701 - 0.417048707955401*I]
```

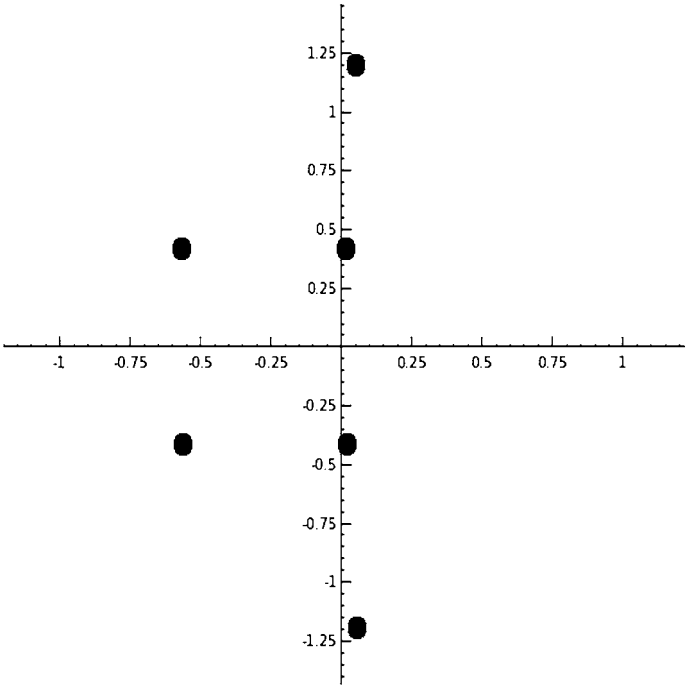


Fig. 4.4 Roots of the Chinen zeta polynomial for a $[8, 4, 3]$ binary code violating the Riemann hypothesis

```
sage: C.gen_mat()
[1 1 0 0 0 0 1 1]
[0 0 1 0 0 1 0 1]
[0 0 0 1 0 1 1 0]
[0 0 0 0 1 1 1 1]
sage: C.chinen_polynomial()
1/7*t^6 + 1/7*t^5 + 39/140*t^4 + 17/70*t^3 + 39/280*t^2 + 1/28*t + 1/56
sage: list_plot([(z[0].real(),z[0].imag()) for z in f.roots()])
```

The last command gives a plot of the roots (see Fig. 4.4).

Chapter 5
Hyperelliptic Curves and Quadratic Residue Codes

For an odd prime p and a nonempty subset $S \subset GF(p)$, consider the hyperelliptic curve X_S defined by

$$y^2 = f_S(x),$$

where $f_S(x) = \prod_{a \in S}(x - a)$. Since the days of E. Artin in the early 1900s, mathematicians have searched for good estimates for the number of points on such curves. In the late 1940s and early 1950s, A. Weil developed good estimates when the genus is small relative to the size of the prime p. When the genus is large compared to p, good estimates are still unknown.

A long-standing problem has been to develop "good" binary linear codes to be used for error-correction. For example, is the Gilbert–Varshamov bound asymptotically exact in the case of binary codes?

This chapter is devoted to explaining a basic link between these two unsolved problems. Using a connection between binary quadratic residue codes and hyperelliptic curves over $GF(p)$, this chapter investigates how coding theory bounds give rise to bounds such as the following example: for all sufficiently large primes p, there exists a subset $S \subset GF(p)$ for which the bound $|X_S(GF(p))| > 1.39p$ holds.

Felipe Voloch [V2] has kindly allowed the authors to include some interesting explicit constructions (which do not use any theory of error-correcting codes) in this chapter (see Sect. 5.8 below). First, he shows the following result.

Theorem 148 (Voloch) *If $p \equiv 1 \pmod 8$, then there exists an effectively computable subset $S \subset GF(p)$ for which the bound $|X_S(GF(p))| > 1.5p$ holds.*

A similar result holds for $p \equiv 3, 7 \pmod 8$. Second, he gives a construction which answers Open Problem 28 in the negative.

Related to the key issue here of associating character sums with weights of codes are the following (incomplete list of) papers: Shokrollahi [Sh2], van der Vlugt [vdV], Schoof and van der Vlugt [SvdV] (see also Schoof [Scf] and van der Geer, Schoof, and van der Vlugt [VSV]), and the early papers by McEliece and Baumert [MB] and McEliece and Rumsey [MR].

D. Joyner, J.-L. Kim, *Selected Unsolved Problems in Coding Theory*,
Applied and Numerical Harmonic Analysis,
DOI 10.1007/978-0-8176-8256-9_5,

We also use the quasi-quadratic residue codes defined below to construct an example of a formally self-dual optimal code whose Duursma zeta function does not satisfy the "Riemann hypothesis."

5.1 Introduction

A long-standing problem has been to develop "good" binary linear codes to be used for error-correction. Another long-standing problem has been, for a smooth curve over a finite field F, to find a nontrivial estimate for the number of F-rational points of X in the case where the size of F is "small" relative to the genus of X.

This chapter investigates in some detail an attack on this problem using a connection between quadratic residue codes and hyperelliptic curves. Codes with this kind of relationship have been investigated in Helleseth [He], Bazzi and Mitter [BM], Voloch [V1], and Helleseth and Voloch [HV]. The rest of this introduction is devoted to explaining in more detail the ideas discussed in later sections.

Let $\mathbb{F} = GF(2)$ be the field with two elements, and $C \subset \mathbb{F}^n$ denote a binary block code of length n.

Denoting the volume of a Hamming sphere of radius r in $\mathbb{F}^n$ by $V(n, r)$, the binary version of the *Gilbert–Varshamov bound* asserts that (given n and d) there is an $[n, k, d]_2$ code C satisfying $k \geq \log_2(\frac{2^n}{V(n,d-1)})$ [HP1].

Open Problem 25 [JV, G2] The binary version of the Gilbert–Varshamov bound is asymptotically exact.

5.2 Points on Hyperelliptic Curves over Finite Fields

For each odd prime $p > 5$, a quasi-quadratic residue code[1] is a linear code of length $2p$. Like the quadratic residue codes, the length and dimension are easy to determine, but the minimum distance is more mysterious. In fact, the weight of each codeword can be explicitly computed in terms of the number of solutions in integers mod p to a certain type of ("hyperelliptic") polynomial equation. To explain the results better, some more notation is needed.

For our purposes, a *hyperelliptic curve* X over $GF(p)$ is a polynomial equation of the form $y^2 = h(x)$, where $h(x)$ is a polynomial with coefficients in $GF(p)$ with distinct roots.[2] The number of solutions to $y^2 = h(x)$ mod p plus the number of "points at infinity" on X will be denoted $|X(GF(p))|$. This quantity can be related to a sum of Legendre characters (see Proposition 153 below), thanks to classical

[1] This code is defined in Sect. 5.5 below.

[2] This overly simplified definition brings to mind the famous Felix Klein quote: "Everyone knows what a curve is, until he has studied enough mathematics to become confused through the countless number of possible exceptions." Please see Tsafsman and Vladut [TV] or Schmidt [Sch] for a rigorous treatment.

work of Artin, Hasse, and Weil. This formula yields good estimates for $|X(GF(p))|$ in many cases (especially where p is large compared to the degree of h). A long-standing problem has been to improve the trivial estimate when p is small compared to the degree of h. It turns out that the work of Tarnanen [T] easily yields some nontrivial information on this problem (see, for example, Lemma 154 below), but the results given here improve upon this.

For each nonempty subset $S \subset GF(p)$, consider the hyperelliptic curve X_S defined by $y^2 = f_S(x)$, where $f_S(x) = \prod_{a \in S}(x - a)$. Let $B(c, p)$ be the statement: *For all subsets* $S \subset GF(p)$, $|X_S(GF(p))| \leq c \cdot p$ *holds.* Note that $B(2, p)$ is trivially true, so the statement $B(2 - \epsilon, p)$ for some fixed $\epsilon > 0$ might not be unreasonable.

Open Problem 26 (Bazzi–Mitter conjecture, [BM]) There is a $c \in (0, 2)$ such that, for an infinite number of primes p, the statement $B(c, p)$ holds.

It is remarkable that these two conjectures (namely, Open Questions 25 and 26) are related. In fact, using quasi-quadratic residue codes, we show that if, for an infinite number of primes p with $p \equiv 1 \pmod 4$, $B(1.77, p)$ holds, then Goppa's conjecture is false. This result, remarkably enough, turns out to be an easy consequence of the quasi-quadratic residue construction given in [BM]. Using long quadratic residue codes,[3] we will remove the condition $p \equiv 1 \pmod 4$ at a cost of slightly weakening the constant 1.77 (see Corollary 164).

In Sect. 5.6 below, the spectrum and Duursma zeta function of these quasi-quadratic residue codes is discussed, and some examples are given (with the help of the software package SAGE [S]). We show that the analog of the Riemann hypothesis for the zeta function of an optimal formally self-dual code is false using the family of codes constructed in Sect. 5.5. The section ends with some intriguing conjectures.

We close this introduction with a few open questions which, on the basis of this result, seem natural.

Open Problem 27 For each prime $p > 5$, is there an effectively computable subset $S \subset GF(p)$ such that $|X_S(GF(p))|$ is "large"?

Here "large" is left vague, but what is intended is some quantity which is unusual. By Weil's estimate (valid for "small"-sized subsets S), we could expect about p points to belong to $|X_S(GF(p))|$. Thus "large" could mean, say, $> c \cdot p$, for some fixed $c > 1$.

The next question is a strong version of the Bazzi–Mitter conjecture.

Open Problem 28 Does there exist a $c < 2$ such that, for all sufficiently large p and all $S \subset GF(p)$, we have $|X_S(GF(p))| < c \cdot p$?

In the direction of these questions, for Open Problem 27, a coding theory bound of McEliece–Rumsey–Rodemich–Welsh allows one to establish the following result

[3] These codes will be defined in Sect. 5.7 below.

(see Theorem 164): *There exists a constant p_0 having the following property: if $p > p_0$, then there exists a subset $S \subset GF(p)$ for which the bound $|X_S(GF(p))| > 1.62p$ holds*. Unfortunately, the method of proof gives no clue how to compute p_0 or S. Using the theory of long quadratic-residue codes, we prove the following lower bound (Theorem 174): For all $p > p_0$, there exists a subset $S \subset GF(p)$ for which the bound $|X_S(GF(p))| > 1.39p$ holds. Again, we do not know what p_0 or S is.

5.3 Non-Abelian Group Codes

The following construction generalizes the above example in an abstract way but will be needed later.

Let G be any finite group, and let $\mathbb{F}$ be any finite field.

Here is a very general construction of a code C whose automorphism group contains G.

If x is an indeterminate and $g \in G$, then we let the formal symbol x^g denote "gth power" of x. The group algebra

$$\mathbb{F}[G] = \left\{ \sum_{g \in G} c_g x^g \mid c_g \in \mathbb{F} \right\}$$

is a left G-module under the action

$$\lambda(g)\left(x^h\right) = x^{gh}, \quad g, h \in G.$$

(Note: $\lambda(g_1)\lambda(g_2)(x^h) = \lambda(g_1)x^{g_2 h} = x^{g_1 g_2 h} = \lambda(g_1 g_2)(x^h)$ for $g_1, g_2, h \in G$.) Therefore, λ defines an action of G on $\mathbb{F}[G]$ called the *regular representation*. Let n denote the dimension of $\mathbb{F}[G]$ (so $n = |G|$ is simply the size of G since the "coordinates" of an element of $\mathbb{F}[G]$ are indexed by G).

Now, pick any element $a \in \mathbb{F}[G]$ and consider the G-orbit of a,

$$G \cdot a = \left\{ \lambda(g)(a) \mid g \in G \right\}.$$

If $a = \sum_{h \in G} c_h x^h$, then $\lambda(g)(a) = \sum_{h \in G} c_h x^{gh} = \sum_{h \in G} c_{g^{-1}h} x^h$. Finally, let C be the vector subspace spanned by $G \cdot a$:

$$C = \mathrm{Span}\left(\left\{\lambda(g)(a) | g \in G\right\}\right) = \mathrm{Span}\left(\left\{ \sum_{h \in G} c_{g^{-1}h} x^h \mid g \in G \right\}\right).$$

In this case, G acts on C by permuting coordinates via the left action of G on itself, so $G \subset \mathrm{Aut}(C)$. More generally, one may take C to be any G-submodule of $\mathbb{F}[G]$.

5.4 Cyclotomic Arithmetic mod 2

Quadratic residue codes were introduced in Sect. 1.6.3 above. In this and the following sections, we focus on the binary case.

Let $R = \mathbb{F}[x]/(x^p - 1)$, and let $r_S \in R$ be the polynomial

$$r_S(x) = \sum_{i \in S} x^i,$$

where $S \subseteq GF(p)$. By convention, $r_S = 0$ if $S = \emptyset$. We define the *weight* of r_S, denoted $\mathrm{wt}(r_S)$, to be the cardinality $|S|$. (In other words, identify in the obvious way each r_S with an element of $\mathbb{F}^p$ and define the weight of r_S to be the Hamming weight of the associated vector.) For the set Q of quadratic residues in $GF(p)^\times$ and the set N of nonquadratic residues in $GF(p)^\times$, we have $\mathrm{wt}(r_Q) = \mathrm{wt}(r_N) = (p-1)/2$. Note that $r_S^2 = r_{2S}$, where $2S$ is the set of elements $2s \in GF(p)$ for $s \in S$. Using this fact and the quadratic reciprocity law, one can easily show that the following are equivalent:

- $r_Q^2 = r_Q$,
- $2 \in Q$,
- $p \equiv \pm 1 \pmod 8$.

Moreover, if $2 \in N$, then $r_Q^2 = r_N$.

Let S, S_1, S_2, S_1' denote subsets of $GF(p)$ with $S_1 \cap S_1' = \emptyset$, and let $S^c = GF(p) - S$ denote the complement. For $a \in GF(p)$, let

$$H(S_1, S_2, a) = \big\{(s_1, s_2) \in S_1 \times S_2 \,|\, s_1 + s_2 \equiv a \pmod p\big\}.$$

In particular,

- $H(S_1, S_2, a) = H(S_2, S_1, a)$,
- there is a natural bijection $H(GF(p), S, a) \cong S$,
- if $S_1 \cap S_1' = \emptyset$, then $H(S_1, S_2, a) + H(S_1', S_2, a) = H(S_1 + S_1', S_2, a)$.

Let

$$h(S_1, S_2, a) = \big|H(S_1, S_2, a)\big| \pmod 2.$$

Adding $|H(S_1, S_2, a)| + |H(S_1^c, S_2, a)| = |S_2|$ to $|H(S_1^c, S_2^c, a)| + |H(S_1^c, S_2, a)| = |S_1^c|$, we obtain

$$h(S_1, S_2, a) \equiv h\big(S_1^c, S_2^c, a\big) + \big|S_1^c\big| + |S_2| \pmod 2. \tag{5.4.1}$$

From the definition of r_S we have

$$r_{S_1}(x) r_{S_2}(x) = \sum_{a \in GF(p)} h(S_1, S_2, a) x^a$$

in the ring R. Let $* : R \to R$ denote the involution defined by $(r_S)^* = r_{S^c} = r_S + r_{GF(p)}$. We shall see below that this is not an algebra involution.

Lemma 149 *For all $S_1, S_2 \subset GF(p)$, we have*:

- *$|S_1|$ odd, $|S_2|$ even: $r_{S_1} r_{S_2} = r_{S_1}^* r_{S_2}^*$ has even weight*;
- *$|S_1|$ even, $|S_2|$ even: $(r_{S_1} r_{S_2})^* = r_{S_1}^* r_{S_2}^*$ has even weight*;

- $|S_1|$ *even*, $|S_2|$ *odd*: $r_{S_1} r_{S_2} = r^*_{S_1} r^*_{S_2}$ *has even weight*;
- $|S_1|$ *odd*, $|S_2|$ *odd*: $(r_{S_1} r_{S_2})^* = r^*_{S_1} r^*_{S_2}$ *has odd weight*.

This lemma follows from the discussion above by a straightforward argument.

Note that $R_{\text{even}} = \{r_S \,|\, |S| \text{ even}\}$ is a subring of R, and, by the previous lemma, $*$ is an algebra involution on R_{even}.

5.5 Quasi-quadratic Residue Codes

These are some observations on the interesting paper by Bazzi and Mitter [BM]. We shall need to remove the assumption $p \equiv 3 \pmod 8$ (which they make in their paper) below.

If $S \subseteq GF(p)$, let $f_S(x) = \prod_{a \in S}(x - a) \in GF(p)[x]$. Let $\chi = (\frac{\cdot}{p})$ be the quadratic residue character, which is equal to 1 on the quadratic residues $Q \subset GF(p)^\times$, -1 on the quadratic nonresidues $N \subset GF(p)^\times$, and is equal to 0 at $0 \in GF(p)$.

Define

$$C_{NQ} = \left\{(r_N r_S, r_Q r_S) \,|\, S \subseteq GF(p)\right\},$$

where N, Q are as above. (We identify in the obvious way each pair $(r_N r_S, r_Q r_S)$ with an element of $\mathbb{F}^{2p}$. In particular, when S is the empty set, $(r_N r_S, r_Q r_S)$ is associated with the zero vector in $\mathbb{F}^{2p}$.) We call this a *quasi-quadratic residue code*. These are binary linear codes of length $2p$ and dimension

$$k = \begin{cases} p & \text{if } p \equiv 3 \pmod 4, \\ p-1 & \text{if } p \equiv 1 \pmod 4. \end{cases}$$

This code has no codewords of odd weight, for parity reasons, by Lemma 149.

Remark 18 If $p \equiv \pm 1 \pmod 8$, then C_{NQ} "contains" a binary quadratic residue code. For such primes p, the minimum distance satisfies the well-known square-root lower bound $d \geq \sqrt{p}$.

Based on computations using SAGE, the following statement is likely to be true [Jo1].

Proposition 150 1. *For* $p \equiv 1 \pmod 4$, *the associated quasi-quadratic residue code and its dual satisfy* $C_{NQ} \oplus C^\perp_{NQ} = \mathbb{F}^{2p}$, *where* $\oplus$ *stands for the direct product* (*so, in particular,* $C_{NQ} \cap C^\perp_{NQ} = \{\mathbf{0}\}$).

2. *If* $p \equiv 3 \pmod 4$, *then the associated quasi-quadratic residue code is self-dual*: $C^\perp_{NQ} = C_{NQ}$.

In 2008 Robin Chapman [Cha] and Maosheng Xiong reported to the first author that Proposition 150 (stated as a conjecture in [Jo1]) is true. As their proofs

are similar, we describe Chapman's algebraic proof as follows. We will also give a combinatorial proof for the second part (due to the second author).

Proof of Proposition 150 Define a ring automorphism $f \to f^-$ of R by $f^-(x) = f(x^{-1})$, where x^{-1} denotes x^{p-1}. We also define an $\mathbb{F}$-linear map $\varepsilon : R \to \mathbb{F}$ by

$$\varepsilon\left(\sum_{j=0}^{p-1} a_j x^j\right) = a_0.$$

Then this gives an $\mathbb{F}$-bilinear pairing on R defined by

$$\langle f, g\rangle = \varepsilon\big(fg^-\big).$$

This pairing corresponds to the standard inner product on $\mathbb{F}^p$ since for $f = \sum_{j=0}^{p-1} a_j x^j = (a_0, \ldots, a_{p-1})$ and $g = \sum_{j=0}^{p-1} b_j x^j = (b_0, \ldots, b_{p-1})$, $\varepsilon(fg^-) = \sum_{j=0}^{p-1} a_j b_j = \langle f, g\rangle$. By identifying $\mathbb{F}^{2p}$ with R^2, one can give R^2 the pairing $\langle (f_1, f_2), (g_1, g_2)\rangle = \langle f_1, g_1\rangle + \langle f_2, g_2\rangle$, which corresponds to the standard inner product on $\mathbb{F}^{2p}$.

Let $t = t(x) := \sum_{j=0}^{p-1} x^j = 1 + r_N + r_Q$. We first show that if $p \equiv 1 \pmod 4$, then $C_{NQ} \oplus C_{NQ}^{\perp} = \mathbb{F}^{2p}$. It is enough to show that $C_{NQ} \cap C_{NQ}^{\perp} = \{0\}$. To prove this, suppose that $(f_{rN}, f_{rQ}) \in C_{NQ} \cap C_{NQ}^{\perp}$. Then, for each $g \in R$,

$$\begin{aligned}
0 &= \big\langle (fr_N, fr_Q), (gr_N, gr_Q)\big\rangle = \varepsilon\big(fg^-\big(r_N r_N^- + r_Q r_Q^-\big)\big) \\
&= \varepsilon\big(fg^-\big(r_N^2 + r_Q^2\big)\big) \text{ as } r_N^- = r_N, r_Q^- = r_Q \text{ when } p \equiv 1 \pmod 4 \\
&= \varepsilon\big(fg^-(r_N + r_Q)^2\big) \\
&= \varepsilon\big(fg^-(t-1)^2\big) = \epsilon\big(fg^-(t-1)\big) \\
&= \big\langle f(t-1), g\big\rangle.
\end{aligned}$$

As the pairing is nonsingular, $f(t-1) = 0$. Hence $f(x) = f(x)t(x) = f(1)t(x) \in \{0, t(x)\}$. If $f(x) = 0$, then done. Suppose that $f(x) = t(x)$. Then $f(x)r_N(x) = t(x)r_N(x) = r_N(1)t(x) = 0$, where the second equality follows as $t(x)$ is the all-ones vector, and the third equality follows as the weight of $r_N(1)$ is even. By symmetry, $f(x)r_Q(x) = 0$. Therefore $(fr_N, fr_Q) = 0$. Thus $C_{NQ} \cap C_{NQ}^{\perp} = \{0\}$, proving the first part of the conjecture.

Next we show that if $p \equiv 3 \pmod 4$, then C_{NQ} is self-dual. Because C_{NQ} is p-dimensional, it suffices to show that C_{NQ} is self-orthogonal as follows. For any $f, g \in R$,

$$\begin{aligned}
\big\langle (fr_N, fr_Q), (gr_N, gr_Q)\big\rangle &= \varepsilon\big(fg^-\big(r_N r_N^- + r_Q r_Q^-\big)\big) \\
&= \varepsilon\big(fg^-(r_N r_Q + r_Q r_N)\big) \\
&= \varepsilon\big(fg^-(2r_N r_Q)\big) = 0.
\end{aligned}$$

This proves the second part of the conjecture. □

Remark 19 We can show the second part of Proposition 150 in a combinatorial flavor as follows.

Suppose that $p = 4k + 3$ for some $k \geq 0$. Let $\mathcal{N}$ and $\mathcal{Q}$ be the circulant matrices whose first rows are r_N, r_Q, respectively. We note that C_{NQ} has a generator matrix $G(C_{NQ}) = [\mathcal{N} \mid \mathcal{Q}]$ for any odd prime p. It is known that $\mathcal{Q}\mathcal{Q}^T = (k+1)I + kJ$ and that $\mathcal{N}\mathcal{N}^T = (k+1)I + kJ$ (for example, see [Ga]). Thus, C_{NQ} is self-orthogonal as $[\mathcal{N} \mid \mathcal{Q}][\mathcal{N} \mid \mathcal{Q}]^T = \mathcal{N}\mathcal{N}^T + \mathcal{Q}\mathcal{Q}^T = 2(k+1)I + 2kJ \equiv 0 \pmod 2$. Since C_{NQ} has dimension p, C_{NQ} is self-dual.

The self-dual binary codes have useful upper bounds on their minimum distance (for example, the Sloane–Mallows bound, Theorem 9.3.5 in [HP1]). Combining this with the lower bound mentioned above, we have the following result.

Lemma 151 *If* $p \equiv 3 \pmod 4$, *then*

$$d \leq 4 \cdot [p/12] + 6.$$

If $p \equiv -1 \pmod 8$, *then*

$$\sqrt{p} \leq d \leq 4 \cdot [p/12] + 6.$$

Note that these upper bounds (in the cases they are valid) are better than the asymptotic bounds of McEliece–Rumsey–Rodemich–Welsh for rate $1/2$ codes.

Example 152 The following computations were done with the help of SAGE. When $p = 5$, C_{NQ} has the weight distribution

$$[1, 0, 0, 0, 5, 0, 10, 0, 0, 0, 0, 0].$$

When $p = 7$, C_{NQ} has the weight distribution

$$[1, 0, 0, 0, 14, 0, 49, 0, 49, 0, 14, 0, 0, 0, 1].$$

When $p = 11$, C_{NQ} has the weight distribution

$$[1, 0, 0, 0, 0, 0, 77, 0, 330, 0, 616, 0, 616, 0, 330, 0, 77, 0, 0, 0, 0, 0, 1].$$

When $p = 13$, C_{NQ} has the weight distribution

$$[1, 0, 0, 0, 0, 0, 0, 0, 273, 0, 598, 0, 1105, 0, 1300, 0, 598, 0, 182, 0, 39, 0, 0, 0, 0, 0, 0].$$

The following well-known result[4] will be used to estimate the weights of codewords of quasi-quadratic residue codes.

[4] See, for example, Weil [W] or Schmidt [Sch], Lemma 2.11.2.

Proposition 153 (Artin, Hasse, Weil) *Assume that $S \subset GF(p)$ is nonempty.*

- $|S|$ *even*:

$$\sum_{a \in GF(p)} \chi\big(f_S(a)\big) = -p - 2 + \big|X_S\big(GF(p)\big)\big|.$$

- $|S|$ *odd*:

$$\sum_{a \in GF(p)} \chi\big(f_S(a)\big) = -p - 1 + \big|X_S\big(GF(p)\big)\big|.$$

- $|S|$ *odd*: *The genus of the* (*smooth projective model of the*) *curve* $y^2 = f_S(x)$ *is* $g = \frac{|S|-1}{2}$, *and*

$$\left| \sum_{a \in GF(p)} \chi\big(f_S(a)\big) \right| \le \big(|S| - 1\big) p^{1/2} + 1.$$

- $|S|$ *even*: *The genus of the* (*smooth projective model of the*) *curve* $y^2 = f_S(x)$ *is* $g = \frac{|S|-2}{2}$, *and*

$$\left| \sum_{a \in GF(p)} \chi\big(f_S(a)\big) \right| \le \big(|S| - 2\big) p^{1/2} + 1.$$

Obviously, the last two estimates are only nontrivial for S "small" (e.g., $|S| < p^{1/2}$).

Lemma 154 (Tarnanen [T], Theorem 1) *Fix τ, $0.39 < \tau < 1$. For all sufficiently large p, the following statement is false: For all subsets $S \subset GF(p)$ with $|S| \le \tau p$, we have $0.42p < |X_S(GF(p))| < 1.42p$.*

Remark 20 (1) Here the meaning of "sufficiently large" is hard to make precise. The results of Tarnanen are actually asymptotic (as $p \to \infty$), so we can simply say that the negation of part (1) of this lemma contradicts Theorem 1 in [T].

(2) This lemma does not seem to imply "$B(1.42, p)$ is false for sufficiently large p" (so Theorem 174 below is a new result), though it would if the condition $0.42p < |X_S(GF(p))|$ could be eliminated. Also of interest is the statement about character sums in Theorem 1 of Stepanov [St2].

Proof This is an immediate consequence of the proposition above and Theorem 1 in [T]. □

Lemma 155 (Bazzi–Mitter [BM], Proposition 3.3) *Assume that 2 and -1 are quadratic nonresidues mod p (i.e., $p \equiv 3 \pmod 8$).*

If $\mathbf{c} = (r_N r_S, r_Q r_S)$ is a nonzero codeword of the $[2p, p]$ binary code C_{NQ}, then the weight of this codeword can be expressed in terms of a character sum as

$$\mathrm{wt}(\mathbf{c}) = p - \sum_{a \in GF(p)} \chi\big(f_S(a)\big)$$

if $|S|$ is even and

$$\mathrm{wt}(\mathbf{c}) = p + \sum_{a \in GF(p)} \chi\big(f_{S^c}(a)\big)$$

if $|S|$ is odd.

In fact, looking carefully at their proof, one finds the following result.

Proposition 156 *Let $\mathbf{c} = (r_N r_S, r_Q r_S)$ be a nonzero codeword of C_{NQ}.*

(a) *If $|S|$ is even,*

$$\mathrm{wt}(\mathbf{c}) = p - \sum_{a \in GF(p)} \chi\big(f_S(a)\big) = 2p + 2 - \big|X_S\big(GF(p)\big)\big|.$$

(b) *If $|S|$ is odd and $p \equiv 1 \pmod 4$, then the weight is*

$$\mathrm{wt}(\mathbf{c}) = p - \sum_{a \in GF(p)} \chi\big(f_{S^c}(a)\big) = 2p + 2 - \big|X_{S^c}\big(GF(p)\big)\big|.$$

(c) *If $|S|$ is odd and $p \equiv 3 \pmod 4$, then*

$$\mathrm{wt}(\mathbf{c}) = p + \sum_{a \in GF(p)} \chi\big(f_{S^c}(a)\big) = \big|X_{S^c}\big(GF(p)\big)\big| - 2.$$

Proof If $A, B \subseteq GF(p)$, then the discussion in Sect. 5.4 implies

$$\mathrm{wt}(r_A r_B) = \sum_{k \in GF(p)} \mathrm{parity}\big|A \cap (k - B)\big|, \tag{5.5.1}$$

where $k - B = \{k - b | b \in B\}$, and $\mathrm{parity}(x) = 1$ if x is an odd integer and $= 0$ otherwise. Let $S \subseteq GF(p)$. Then we have

$$p - \mathrm{wt}(r_Q r_S) - \mathrm{wt}(r_N r_S) = \sum_{a \in GF(p)} \big(1 - \mathrm{parity}\big|Q \cap (a - S)\big| - \mathrm{parity}\big|N \cap (a - S)\big|\big).$$

Let

$$T_a(S) = 1 - \mathrm{parity}\big|Q \cap (a - S)\big| - \mathrm{parity}\big|N \cap (a - S)\big|.$$

Case 1. If $|S|$ is even and $a \in S$, then $0 \in a - S$, so $|Q \cap (a - S)|$ odd implies that $|N \cap (a - S)|$ is even, since 0 is not included in $Q \cap (a - S)$ or $N \cap (a - S)$. Likewise, $|Q \cap (a - S)|$ even implies that $|N \cap (a - S)|$ is odd. Therefore, $T_a(S) = 0$.

Case 2. If $|S|$ is even and $a \notin S$, then parity $|Q \cap (a - S)| = \mathrm{parity}|N \cap (a - S)|$. If $|Q \cap (a - S)|$ is even, then $T_a(S) = 1$, and if $|Q \cap (a - S)|$ is odd, then $T_a(S) = -1$.

Case 3. $|S|$ is odd. We claim that $(a - S)^c = a - S^c$. (Proof: Let $s \in S$ and $\bar{s} \in S^c$. Then $a - s = a - \bar{s} \implies s = \bar{s}$, which is obviously a contradiction. Therefore,

$(a - S) \cap (a - S^c) = \emptyset$, so $(a - S)^c \supseteq (a - S^c)$. Replace S by S^c to prove the claim.) Also note that

$$\big(Q \cap (a - S)\big) \sqcup \big(Q \cap \big(a - S^c\big)\big) = GF(p) \cap Q = Q$$

has $|Q| = \frac{p-1}{2}$ elements ($\sqcup$ denotes disjoint union). So

$$\text{parity}\big|Q \cap (a - S)\big| = \text{parity}\big|Q \cap \big(a - S^c\big)\big|$$

if and only if $|Q|$ is even, and

$$\text{parity}\big|Q \cap (a - S)\big| \neq \text{parity}\big|Q \cap \big(a - S^c\big)\big|$$

if and only if $|Q|$ is odd.

Conclusion.

$$|S| \text{ even:} \quad T_a(S) = \prod_{x \in a - S} \left(\frac{x}{p}\right);$$

$$|S| \text{ odd and } p \equiv 3 \pmod 4\text{:} \quad T_a(S) = -T_a\big(S^c\big);$$

$$|S| \text{ odd and } p \equiv 1 \pmod 4\text{:} \quad T_a(S) = T_a\big(S^c\big).$$

The relation between wt(**c**) and the character sum follows from this. For the remaining part of the equation, use Proposition 153. □

Remark 21 It can be shown, using the coding-theoretic results above, that if $p \equiv -1 \pmod 8$, then (for nonempty S) $X_S(GF(p))$ contains at least $\sqrt{p} + 1$ points. This also follows from Weil's estimate, but since the proof is short, it is given here.

Part (c) of Proposition 156 gives that if $p \equiv -1 \pmod 8$ and $|S|$ is odd, then $X_S(GF(p))$ contains at least $\sqrt{p} + 2$ points. If $|S|$ is even, then perform the substitution $x = a + 1/\overline{x}$, $y = \overline{y}/\overline{x}^{|S|}$ on the equation $y^2 = f_S(x)$. This creates a hyperelliptic curve X in $(\overline{x}, \overline{y})$ for which $|X(GF(p))| = |X_S(GF(p))|$ and $X \cong X_{S'}$, where $|S'| = |S| - 1$ is odd. Now apply part (c) of the above proposition and Remark 18 to $X_{S'}$. □

Remark 22 If $|S| = 2$ or $|S| = 3$, then, thanks to Wage [Wa], more can be said about the character sums above.

- If $|S| = 2$, then $\sum_a \chi(f_S(a))$ can be computed explicitly (it is "usually" equal to -1; see Proposition 1 in Wage's paper).
- If $|S| = 3$, then $\sum_a \chi(f_S(a))$ can be expressed in terms of a hypergeometric function over the finite field $GF(p)$,

$$ {}_2F_1(t) = \frac{\psi(-1)}{p} \sum_{x \in GF(p)} \psi(x)\psi(1 - x)\psi(1 - tx),$$

where $\psi(x) = \left(\frac{x}{p}\right)$ is the Legendre symbol. See Proposition 2 in that paper for further details.

It has already been observed that the following fact is true. Since its proof using basic facts about hyperelliptic curves is so short, it is included here.

Corollary 157 *C_{NQ} is an even weight code.*

Proof Since p is odd, $1 \neq -1$ in $GF(p)$, so every affine point in $X_S(GF(p))$ occurs as an element of a pair of solutions of $y^2 = f_S(x)$. There are two points at infinity (if ramified, it is counted with multiplicity two), so in general $|X_S(GF(p))|$ is even. The formulas for the weight of a codeword in the above proposition imply that every codeword has even weight. □

As a consequence of this proposition and Lemma 151, we have the following result.

Corollary 158 *If $p \equiv 3 \pmod 4$, then* $\max_S |X_S(GF(p))| > \frac{5}{3}p - 4$.

Example 159 The following examples were computed with the help of SAGE.

If $p = 11$ and $S = \{1, 2, 3, 4\}$, then

$$\big(r_S(x)r_N(x), r_S(x)r_Q(x)\big) = \big(x^{10} + x^9 + x^7 + x^6 + x^5 + x^4 + x^2 + 1, \\ x^{10} + x^9 + x^7 + x^6 + x^5 + x^3 + x + 1\big)$$

corresponds to the codeword $(1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1)$ of weight 16. An explicit computation shows that the value of the character sum $\sum_{a \in GF(11)} \chi(f_S(a))$ is -5, as expected.

If $p = 11$ and $S = \{1, 2, 3\}$, then

$$\big(r_S(x)r_N(x), r_S(x)r_Q(x)\big) = \big(x^9 + x^7 + x^5 + x^4 + x^3 + x^2 + x, \\ x^{10} + x^8 + x^6 + x^3 + x^2 + x + 1\big)$$

corresponds to the codeword $(0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0, 1)$ of weight 14. An explicit computation shows that the value of the character sum $\sum_{a \in GF(11)} \chi(f_{S^c}(a))$ is 3, as predicted.

Recall that $B(c, p)$ is the following statement: $|X_S(GF(p))| \leq c \cdot p$ for all $S \subset GF(p)$.

Theorem 160 (Bazzi–Mitter) *Fix $c \in (0, 2)$. If $B(c, p)$ holds for infinitely many p with $p \equiv 1 \pmod 4$, then there exists an infinite family of binary codes with asymptotic rate $R = 1/2$ and relative distance $\delta \geq 1 - \frac{c}{2}$.*

This is an easy consequence of the above proposition and is essentially in [BM] (though they assume that $p \equiv 3 \pmod 8$).

Theorem 161 *If $B(1.77, p)$ is true for infinitely many primes p with $p \equiv 1 \pmod 4$, then Goppa's conjecture is false.*

Proof Recall that Goppa's conjecture is that the binary asymptotic Gilbert–Varshamov bound is best possible for any family of binary codes. The asymptotic GV bound states that the rate R is greater than or equal to $1 - H_2(\delta)$, where

$$H_q(\delta) = \delta \cdot \log_q(q-1) - \delta \log_q(\delta) - (1-\delta)\log_q(1-\delta)$$

is the entropy function (for a q-ary channel). Therefore, according to Goppa's conjecture, if $R = \frac{1}{2}$ (and $q = 2$), then the best possible δ is $\delta_0 = 0.11$. Assume that $p \equiv 1 \pmod 4$. Goppa's conjecture implies that the minimum distance of our quasi-quadratic residue code with rate $R = \frac{1}{2}$ satisfies $d < \delta_0 \cdot 2p = 0.22p$ for sufficiently large p. Recall that the weight of a codeword in this quasi-quadratic residue code is given by Proposition 156. $B(1.77, p)$ (with $p \equiv 1 \pmod 4$) implies (for all $S \subset GF(p)$) $\mathrm{wt}((r_S r_N, r_S r_Q)) \geq 2p - |X_S(GF(p))| \geq 0.23p$. In other words, for $p \equiv 1 \pmod 4$, all nonzero codewords have weight at least $0.23p$. This contradicts the estimate above. □

Theorem 162 (First asymptotic McEliece–Rumsey–Rodemich–Welsh bound, [HP1] Theorem 2.10.6) *The rate $R = k/n$ of any $[n, k, d]_2$-code is less than or equal to*

$$h(\delta) = H_2\left(\frac{1}{2} - \sqrt{\delta(1-\delta)}\right),$$

where $\delta = d/n$.

For brevity, this result will be referred to as the MRRW bound.

Using the same arguments in the above proof and the MRRW bound, we prove the following unconditional result.

Theorem 163 *For all sufficiently large primes p for which $p \equiv 1 \pmod 4$, the statement $B(1.62, p)$ is false.*

Proof If a prime p satisfies $B(1.62, p)$, then we shall call it "admissible." We show that the statement "$B(1.62, p)$ holds for all sufficiently large primes p for which $p \equiv 1 \pmod 4$" contradicts the MRRW bound.

Theorem 162 and the fact that $R = \frac{1}{2}$ for our quasi-quadratic residue codes (with $p \equiv 1 \pmod 4$) imply $\delta \leq \delta_0 = h^{-1}(1/2) \cong 0.187$. Therefore, for all large p (admissible or not), $d \leq \delta_0 \cdot 2p$. On the other hand, if p is admissible and $|X_S(GF(p))| \leq c \cdot p$ (where $c = 1.62$), then by the above argument, $d \geq 2 \cdot (p - \frac{c}{2}p)$. Together, we obtain $1 - \frac{c}{2} \leq \delta_0$, so $c \geq 2 \cdot (1 - h^{-1}(1/2)) \cong 1.626$. This is a contradiction. □

Corollary 164 *There is a constant p_0 (ineffectively computable) having the following property: if $p > p_0$, then there is a subset $S \subset GF(p)$ for which the bound $|X_S(GF(p))| > 1.62p$ holds.*

5.6 Weight Distributions

As we saw in the previous chapter, associated to a linear code C over $GF(q)$ there is a zeta function $Z = Z_C$ of the form

$$Z(T) = \frac{P(T)}{(1-T)(1-qT)},$$

where $P(T)$ is a polynomial of degree $n + 2 - d - d^{\perp}$ which only depends on C through its weight enumerator polynomial (here d is the minimum distance of C, and $d^{\perp}$ is the minimum distance of its dual code $C^{\perp}$; we assume that $d \geq 2$ and $d^{\perp} \geq 2$). We also defined (for formally self-dual codes C) the *Riemann hypothesis* to be the statement that all the zeros occur on the "critical circle."

Example 165 The following computations were done with the help of SAGE. If $p = 7$, then the [14, 7, 4] (self-dual) code C_{NQ} has the "zeta polynomial"

$$P(T) = \frac{2}{143} + \frac{4}{143}T + \frac{19}{429}T^2 + \frac{28}{429}T^3 + \frac{40}{429}T^4 + \frac{56}{429}T^5 + \frac{76}{429}T^6 + \frac{32}{143}T^7 + \frac{32}{143}T^8.$$

It can be checked that all the roots ρ of Z_C have $|\rho| = 1/\sqrt{2}$, thus verifying the Riemann hypothesis in this case.

It would be interesting to know if the Duursma zeta function $Z(T)$ of C_{NQ} for $p \equiv 3 \pmod 4$ always satisfies the Riemann hypothesis.

Proposition 166 *If $p \equiv 1 \pmod 4$, then the code C' spanned by C_{NQ} and the all-ones codeword* (*i.e., the smallest code containing C_{NQ} and all its complementary codewords*) *is a formally self-dual code of dimension p. Moreover, if $A = [A_0, A_1, \ldots, A_n]$ denotes the weight distribution vector of C_{NQ}, then the weight distribution vector of C' is $A + A^*$, where $A^* = [A_n, \ldots, A_1, A_0]$.*

Just like Proposition 150, R. Chapman and M. Xiong showed independently that Proposition 166 (which was also a conjecture in [Jo1]) is true. We give Chapman's algebraic proof first and then our combinatorial proof.

Proof of Proposition 166 (Algebraic proof) Let $\mathbf{1}_{2p}$ be the all-ones vector of length $2p$. The second part follows as $C' = C_{NQ} \cup (\mathbf{1}_{2p} + C_{NQ})$. Hence we only show the first part, i.e., if $p \equiv 1 \pmod 4$, then C' is formally self-dual. In what follows, we show that $C'^{\perp} = C_{QN} + \mathbb{F}(t(x), t(x))$, where $C_{QN} = \{(r_Q r_S, r_N r_S) \mid S \subseteq GF(p)\}$. This will imply that $C'^{\perp}$ is equivalent to C', that is, C' is isodual, implying that C' is formally self-dual. It is easy to see that $(t(x), t(x)) \notin C_{NQ}$ since $t(x)$ has odd weight and $f r_Q$ is even for any $f \in R$ as r_Q has even weight. This implies that

C' has dimension p. It suffices to show that C' is orthogonal to $C_{QN}+\mathbb{F}(t(x),t(x))$ as the latter has dimension p too. Note that $(t(x),t(x))$ is orthogonal to C_{NQ} since

$$\begin{aligned}\langle (fr_N, fr_Q),(t,t)\rangle &= \varepsilon\big(ft^-(r_N+r_Q)\big)\\ &= \varepsilon\big(ft(t-1)\big)=\varepsilon\big(f\big(t^2-t\big)\big)=\varepsilon(f\cdot 0)=0.\end{aligned}$$

In a similar manner, $(t(x),t(x))$ is orthogonal to Q_{QN}. Finally, C_{NQ} and C_{QN} are orthogonal since for any $f,g\in R$,

$$\begin{aligned}\langle (fr_N, fr_Q),(gr_Q, gr_N)\rangle &= \varepsilon\big(fg^-\big(r_Nr_Q^- + r_Qr_N^-\big)\big)\\ &= \varepsilon\big(fg^-(r_Nr_Q+r_Qr_N)\big)=\varepsilon\big(fg^-(2r_Nr_Q)\big)=0.\end{aligned}$$

This completes the proof of Proposition 166.

(Combinatorial proof) Note that C' has a generator matrix $\left[\begin{smallmatrix}\mathcal{N} & \mathcal{Q}\\ \mathbf{1}_p & \mathbf{1}_p\end{smallmatrix}\right]$. We show below that its dual $C'^{\perp}$ has a generator matrix $\left[\begin{smallmatrix}\mathcal{Q} & \mathcal{N}\\ \mathbf{1}_p & \mathbf{1}_p\end{smallmatrix}\right]$. This will imply that C' is isodual and hence formally self-dual.

Let $p\equiv 1 \pmod 4$. We have seen above that $\mathbf{1}_{2p}$ is not in C_{NQ} since $\mathbf{1}_p$ is not in the binary code generated by $\mathcal{N}$, each row of which has even weight. Therefore, $\mathrm{Rank}\left(\left[\begin{smallmatrix}\mathcal{N} & \mathcal{Q}\\ \mathbf{1} & \mathbf{1}\end{smallmatrix}\right]\right)=p$. Further, it is easy to see that

$$\begin{aligned}[\mathcal{N}\,|\,\mathcal{Q}][\mathcal{Q}\,|\,\mathcal{N}]^T &= \mathcal{N}\mathcal{Q}^T+\mathcal{Q}\mathcal{N}^T\\ &= \mathcal{N}\mathcal{Q}+\mathcal{Q}\mathcal{N}=\mathcal{N}\mathcal{Q}+\mathcal{N}\mathcal{Q}\equiv 0 \pmod 2\end{aligned}$$

since $\mathcal{Q}^T=\mathcal{Q}$, $\mathcal{N}^T=\mathcal{N}$, and $\mathcal{Q}\mathcal{N}=\mathcal{N}\mathcal{Q}$ [Ga] if $q\equiv 1 \pmod 4$. Clearly $\mathbf{1}_{2p}$ is orthogonal to $[\mathcal{N}\,|\,\mathcal{Q}]$ and $[\mathcal{Q}\,|\,\mathcal{N}]$. Therefore the dual of C' has a generator matrix $\left[\begin{smallmatrix}\mathcal{Q} & \mathcal{N}\\ \mathbf{1}_p & \mathbf{1}_p\end{smallmatrix}\right]$. This completes the proof of Proposition 166. □

Recall that a self-dual code is called "extremal" if its minimum distance satisfies the Sloane–Mallows bound [D3] and "optimal" if its minimum distance is maximal among all such linear codes of that length and dimension. As noted above, the Duursma zeta function has been conjectured to satisfy the Riemann hypothesis for all extremal self-dual codes C. The example below shows that "extremal self-dual" cannot be replaced by "optimal formally self-dual." This example also shows that the Riemann hypothesis is not valid in general for these "extended quasi-quadratic residue codes."

Example 167 If $p=13$, then C' is a $[26,13,6]$ code with weight distribution

$$[1,0,0,0,0,0,39,0,455,0,1196,0,2405,0,2405,0,1196,0,455,0,39,0,0,0,0,0,1].$$

This is (by coding theory tables, as included in SAGE) an optimal, formally self-dual code. This code C' has the zeta polynomial

$$\begin{aligned} P(T) = \frac{3}{17710} + \frac{6}{8855}T + \frac{611}{336490}T^2 + \frac{9}{2185}T^3 + \frac{3441}{408595}T^4 + \frac{6448}{408595}T^5 \\ + \frac{44499}{1634380}T^6 + \frac{22539}{520030}T^7 + \frac{66303}{1040060}T^8 + \frac{22539}{260015}T^9 + \frac{44499}{408595}T^{10} \\ + \frac{51584}{408595}T^{11} + \frac{55056}{408595}T^{12} + \frac{288}{2185}T^{13} + \frac{19552}{168245}T^{14} + \frac{768}{8855}T^{15} \\ + \frac{384}{8855}T^{16}. \end{aligned}$$

Using SAGE, it can be checked that only 8 of the 12 zeros of this function have absolute value $1/\sqrt{2}$.

5.7 Long Quadratic Residue Codes

We now introduce a new code, constructed similarly to the quasi-quadratic residue codes discussed above:

$$C = \left\{ \left(r_N r_S, r_Q r_S, r_N r_S^*, r_Q r_S^* \right) \mid S \subseteq GF(p) \right\}.$$

We call this a *long quadratic residue code* and identify it with a subset of $\mathbb{F}^{4p}$. Observe that this code is nonlinear.

For any $S \subseteq GF(p)$, let

$$\mathbf{c}_S = \left(r_N r_S, r_Q r_S, r_N r_S^*, r_Q r_S^* \right),$$

and let

$$v_S = (r_N r_S, r_Q r_S, r_N r_S, r_Q r_S).$$

If $S_1 \Delta S_2$ denotes the symmetric difference between S_1 and S_2, then it is easy to check that

$$\mathbf{c}_{S_1} + \mathbf{c}_{S_2} = v_{S_1 \Delta S_2}. \tag{5.7.1}$$

We now compute the size of C using Lemma 149. We prove the following *claim*: if $p \equiv 3 \pmod 4$, then the map that sends S to the codeword $\mathbf{c}_S$ is injective. This implies $|C| = 2^p$. Suppose not; then there are two subsets $S_1,\ S_2 \subseteq GF(p)$ that are mapped to the same codeword. Subtracting gives $\mathbf{c}_{S_1} - \mathbf{c}_{S_2} = \mathbf{c}_{S_1} + \mathbf{c}_{S_2} = v_{S_1 \Delta S_2}$, and the subset $T = S_1 \Delta S_2$ satisfies $r_Q r_T = r_N r_T = r_Q r_{T^c} = r_N r_{T^c} = 0$. If $|T|$ is even, then $0 = (r_Q + r_N) r_T = (r_{GF(p)} - 1) r_T = r_T$. This forces T to be the empty set, so $S_1 = S_2$. Now if $|T|$ is odd, then similar reasoning implies that T^c is the empty set. Therefore, $S_1 = \emptyset$ and $S_2 = GF(p)$ or vice versa. This proves the claim.

In the case $p \equiv 1 \pmod 4$, we *claim*: $|C| = 2^{p-1}$. Again, suppose that there are two subsets $S_1,\ S_2 \subseteq GF(p)$ that are mapped to the same codeword. Then the subset $T = S_1 \Delta S_2$ satisfies $r_Q r_T = r_N r_T = r_Q r_{T^c} = r_N r_{T^c} = 0$. This implies either $T = \emptyset$ or $T = GF(p)$. Therefore, either $S_1 = S_2$ or $S_1 = S_2^c$.

Combining this discussion with Proposition 153, we have proven the following result.

Theorem 168 *The code C has length $n = 4p$ and has size $M = 2^{p-1}$ if $p \equiv 1 \pmod 4$, and size $M = 2^p$ if $p \equiv 3 \pmod 4$. If $p \equiv 3 \pmod 4$, then the minimum nonzero weight is $2p$, and the minimum distance is at least*

$$d_p = 4p - 2 \max_{S \subset GF(p)} \left| X_S\big(GF(p)\big) \right|.$$

If $p \equiv 1 \pmod 4$, then C is a binary $[4p,\ p-1,\ d_p]$-code.

Remark 23 If $p \equiv 3 \pmod 4$, there is no simple reason to think why the minimum distance should actually be less than the minimum nonzero weight.

Lemma 169 *If $p \equiv 1 \pmod 4$, then*

- $v_S = \mathbf{c}_S$,
- $\mathbf{c}_{S_1} + \mathbf{c}_{S_2} = \mathbf{c}_{S_1 \Delta S_2}$,
- *the code C is isomorphic to the quasi-quadratic residue code C_{NQ}.*

In particular, C is linear and of dimension $p - 1$.

Proof It follows from the proof of Theorem 168 that if $p \equiv 1 \pmod 4$, then $r_N r_{S_1} = r_N r_{S_2}$ and $r_Q r_{S_1} = r_Q r_{S_2}$ if and only if $S_2 = S_1^c$. The lemma follows rather easily as a consequence of this and (5.7.1). □

Assume that $p \equiv 3 \pmod 4$. Let

$$V = \big\{ v_S \mid S \subset GF(p) \big\},$$

and let

$$\overline{C} = C \cup V.$$

Lemma 170 *The code $\overline{C}$ is*

- *the smallest linear subcode of $\mathbb{F}^{4p}$ containing C,*
- *dimension $p + 1$,*
- *minimum distance $\min(d_p, 2p)$.*

By abuse of terminology, we call $\overline{C}$ a *long quadratic residue code*.

Proof The first part follows from (5.7.1). The second part follows from a counting argument (as in the proof of Theorem 168). The third part is a corollary of Theorem 168. □

Recall that

$$\mathrm{wt}(r_N r_S, r_Q r_S) = \begin{cases} p - \sum_{a \in GF(p)} \left(\frac{f_S(a)}{p}\right), & |S| \text{ even (any } p), \\ p - \sum_{a \in GF(p)} \left(\frac{f_{S^c}(a)}{p}\right), & |S| \text{ odd and } p \equiv 1 \pmod 4, \\ p + \sum_{a \in GF(p)} \left(\frac{f_{S^c}(a)}{p}\right), & |S| \text{ odd and } p \equiv 3 \pmod 4, \end{cases}$$

by Proposition 156.

Lemma 171 *For each p, the codeword $\mathbf{c}_S = (r_N r_S, r_Q r_S, r_N r_S^*, r_Q r_S^*)$ of C has weight*

$$\mathrm{wt}(\mathbf{c}_S) = \begin{cases} 2p - 2\sum_{a \in GF(p)} \left(\frac{f_S(a)}{p}\right), & p \equiv 1 \pmod 4, \\ 2p, & p \equiv 3 \pmod 4. \end{cases}$$

In other words, if $p \equiv 3 \pmod 4$, then C is a constant-weight code.

Proof Indeed, Proposition 156 implies that if $p \equiv 1 \pmod 4$, then

$$\begin{aligned} \mathrm{wt}\left(r_N r_S, r_Q r_S, r_N r_S^*, r_Q r_S^*\right) &= \mathrm{wt}(r_N r_S, r_Q r_S) + \mathrm{wt}\left(r_N r_S^*, r_Q r_S^*\right) \\ &= 2 \cdot \mathrm{wt}(r_N r_S, r_Q r_S) \\ &= 2p - 2 \sum_{a \in GF(p)} \left(\frac{f_S(a)}{p}\right); \end{aligned} \tag{5.7.2}$$

if $p \equiv 3 \pmod 4$ and $|S|$ is even, then

$$\begin{aligned} \mathrm{wt}\left(r_N r_S, r_Q r_S, r_N r_S^*, r_Q r_S^*\right) &= \mathrm{wt}(r_N r_S, r_Q r_S) + \mathrm{wt}\left(r_N r_S^*, r_Q r_S^*\right) \\ &= p - \sum_{a \in GF(p)} \left(\frac{f_S(a)}{p}\right) + p + \sum_{a \in GF(p)} \left(\frac{f_S(a)}{p}\right) \\ &= 2p; \end{aligned} \tag{5.7.3}$$

and if $p \equiv 3 \pmod 4$ and $|S|$ is odd, then

$$\begin{aligned} \mathrm{wt}\left(r_N r_S, r_Q r_S, r_N r_S^*, r_Q r_S^*\right) &= \mathrm{wt}(r_N r_S, r_Q r_S) + \mathrm{wt}\left(r_N r_S^*, r_Q r_S^*\right) \\ &= p + \sum_{a \in GF(p)} \left(\frac{f_S(a)}{p}\right) + p - \sum_{a \in GF(p)} \left(\frac{f_S(a)}{p}\right) \\ &= 2p. \end{aligned} \tag{5.7.4}$$

□

5.7.1 Examples

Example 172 The following examples were computed with the help of SAGE. When $p = 11$ and $S = \{1, 2, 3, 4\}$, $\mathbf{c}_S$ corresponds to the codeword

$$(0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1,$$
$$0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0)$$

of weight 22. When $p = 11$ and $S = \{1, 2, 3\}$, $\mathbf{c}_S$ corresponds to the codeword

$$(1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0,$$
$$0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0)$$

of weight 22.

It turns out that Lemma 170 allows us to improve the statement of Theorem 161 in Sect. 5.5. The next subsection is devoted to this goal.

5.7.2 Goppa's Conjecture Revisited

We shall now remove the condition $p \equiv 1 \pmod 4$ in one of the results in Sect. 5.5, at a cost of weakening the constant involved.

Assuming that $B(c, p)$ holds, we have that the minimum distance of $\overline{C}$ is $\geq \min(d_p, 2p) \geq 4p(1 - \frac{c}{2})$ and the information rate is $R = \frac{1}{4} + \frac{1}{4p}$. When $R = 1/4$, Goppa's conjecture gives $\delta = 0.214\ldots$. So Goppa's conjecture will be false if $1 - \frac{c}{2} = 0.215$, or $c = 1.57$. We have the following improvement of Theorem 161.

Theorem 173 *If $B(1.57, p)$ is true for infinitely many primes p, then Goppa's conjecture is false.*

A similar argument (using $h(x)$ and the MRRW bound in place of $1 - H_2(x)$ and the hypothetical Goppa bound) gives the following:

Theorem 174 *$B(1.39, p)$ cannot be true for infinitely many primes p. In other words, for all "sufficiently large" p, we must have $X_S(GF(p)) > 1.39p$ for some $S \subset GF(p)$.*

5.8 Some Results of Voloch

The following unpublished results of F. Voloch have been included with his kind permission.

Lemma 175 (Voloch) *If $p \equiv 1, 3 \pmod 8$, then $|X_Q(GF(p))| = 1.5p + a$, where Q is the set of quadratic residues, and a is a small constant, $-\frac{1}{2} \le a \le \frac{5}{2}$.*

A similar bound holds if X_Q is replaced by X_N and $p \equiv 1, 3 \pmod 8$ is replaced by $p \equiv 7 \pmod 8$ (in which case, 2 is a quadratic residue).

Proof By Proposition 153, we know that if $p \equiv 3 \pmod 8$ (so $|Q|$ is odd), then

$$\sum_{a \in GF(p)} \chi\big(f_Q(a)\big) = -p - 1 + \big|X_Q\big(GF(p)\big)\big|.$$

Similarly, if $p \equiv 1 \pmod 8$ (so $|Q|$ is even), then

$$\sum_{a \in GF(p)} \chi\big(f_Q(a)\big) = -p - 2 + \big|X_Q\big(GF(p)\big)\big|.$$

Since $b^{\frac{p-1}{2}} \equiv \chi(b) \pmod p$, we have

$$x^{\frac{p-1}{2}} - 1 = \prod_{a \in Q} (x - a) = f_Q(x), \qquad x^{\frac{p-1}{2}} + 1 = \prod_{a \in N} (x - a).$$

In particular, for all $n \in N$,

$$f_Q(n) = \prod_{a \in Q} (n - a) = n^{\frac{p-1}{2}} - 1 \equiv -2 \pmod p.$$

Since $p \equiv 1, 3 \pmod 8$, we have $\chi(-2) = 1$, so $\chi(f_Q(n)) = 1$ for all $n \in N$. It follows that $|X_Q(GF(p))| = \frac{3}{2}p + \chi(f_Q(0)) + \frac{1}{2}$ (if $p \equiv 3 \pmod 8$) or $|X_Q(GF(p))| = \frac{3}{2}p + \chi(f_Q(0)) + \frac{3}{2}$ (if $p \equiv 1 \pmod 8$). □

Here is an extension of the idea in the above proof. Fix an integer $\ell > 2$. Assuming that ℓ divides $p - 1$, there are distinct ℓth roots $r_1 = 1, r_2, \ldots, r_\ell$ in $GF(p)$ for which $x^{p-1} - 1 = \prod_{i=1}^{\ell} (x^{\frac{p-1}{\ell}} - r_i)$. Also, $x^{\frac{p-1}{\ell}} - 1 = \prod_{a \in P_\ell} (x - a) = f_{P_\ell}(x)$, where P_ℓ denotes the set of nonzero ℓth powers in $GF(p)$.

Claim *It is possible to find an infinite sequence of primes p satisfying $p \equiv 1 \pmod \ell$ and $\chi(r_i - 1) = 1$ for all $2 \le i \le \ell$ (where χ denotes the Legendre character mod p). If the claim is true, then we will have a lower bound for $|X_{P_\ell}(GF(p))|$ on order of $(2 - \frac{1}{\ell})p$, along the lines above, by Proposition 153.*

Proof of claim It is a well-known fact in algebraic number theory that $p \equiv 1 \pmod \ell$ implies that the prime p splits completely in the cyclotomic field $\mathbb{Q}_\ell$ generated by the ℓth roots of unity in $\mathbb{C}$, denoted $\tilde{r}_1 = 1, \tilde{r}_2, \ldots, \tilde{r}_\ell$. The condition $\chi(r_i - 1) = 1$ means that p splits in the extension of $\mathbb{Q}_\ell$ obtained by adjoining $\sqrt{\tilde{r}_i - 1}$ (here $i = 2, \ldots, \ell$). By Chebotarev's density theorem there exist infinitely many such p, as claimed. □

In fact, there are effective versions which give explicit information on computing such p [LO, Se1]. This, together with the previous lemma, proves the following result.

Theorem 176 (Voloch) *If $\ell \geq 2$ is any fixed integer, then for infinitely many primes p, there exists a subset $S \subset GF(p)$ for which $|X_S(GF(p))| = (2 - \frac{1}{\ell})p + a$, where a is a small constant, $-\frac{1}{2} \leq a \leq \frac{5}{2}$.*

In fact, the primes occurs with a positive (Dirichlet) density, and the set S can be effectively constructed.

Open Problem 29 Recent work of Pippa charters [Char] investigates a generalization of quadratic residue codes called higher power residue codes. Do these theorems have analogs for them?

Chapter 6
Codes from Modular Curves

One of the most interesting class of curves, from the perspective of arithmetical algebraic geometry, are the so-called modular curves. Some of the most remarkable applications of algebraic geometry to coding theory arise from these modular curves. It turns out these algebraic-geometric codes ("AG codes") constructed from modular curves can have parameters which beat the Gilbert–Varshamov lower bound if the ground field is sufficiently large.

Open Problem 30 Find an infinite family of binary linear codes which, asymptotically, beats the Gilbert–Varshamov lower bound or prove that no such family exists.

We shall try to make a more precise statement of this open problem below (see also Open Question 25 above). However, the basic idea is to try to use the theory of algebraic curves over a finite field for improvement of the Gilbert–Varshamov lower bound. Since we do not (yet) really know how to tell if an arbitrarily given code arises as an AG code [PSvW], perhaps all this sophistication can be avoided.

The topic of AG codes is sketched briefly in Huffman and Pless [HP1], Chap. 13. More complete treatments are given in Tsfasman, Vladut, and Nogin [TVN], Tsfasman and Vladut [TV], Stichtenoth [Sti], Moreno [Mo], and Nieddereiter and Xing [NX]. These are all recommended texts for background below.

6.1 An Overview

Modular curves are remarkable for many reasons, one of which is their high degree of symmetry. In other words, there are a large number of different automorphisms of the curve onto itself. When these curves are used to construct codes, those codes can display not only unusually good error-correcting ability but also remarkable symmetry properties. Still, there are aspects of the structure of this symmetry which are still unknown.

Let $N > 5$ be a prime. The modular curve $X(N)$ has a natural action by the finite group $G = PSL(2, N)$, namely the projective special linear group with coefficients

D. Joyner, J.-L. Kim, *Selected Unsolved Problems in Coding Theory*,
Applied and Numerical Harmonic Analysis,
DOI 10.1007/978-0-8176-8256-9_6,

in $GF(N)$. In fact, the quotient $X(N)/G$ is isomorphic to $X(1) \cong \mathbb{P}^1$. If D is a $PSL(2, N)$-invariant divisor on $X(N)$, then there is a natural representation of G on the Riemann–Roch space $L(D)$. In this chapter, we discuss some results about the $PSL(2, N)$-module structure of the Riemann–Roch space $L(D)$ (in the case where N is prime and $N \geq 7$).

If D is nonspecial, then a formula in Borne [Bo] gives

$$\big[L(D)\big] = (1 - g_{X(1)})\big[k[G]\big] + \big[\deg_{eq}(D)\big] - \big[\tilde{\Gamma}_G\big]. \tag{6.1.1}$$

Here $g_{X(1)}$ is the genus of $X(1)$ (which is zero), square brackets denote the equivalence class of a representation of G, $\deg_{eq}(D)$ is the equivariant degree of D, and $\tilde{\Gamma}_G$ is the ramification module (these terms and notions are defined in Sect. 7.5 below, in [JK2], and in Borne's paper [Bo]). This result is mostly stated here for later reference.

The G-module structure of $L(D)$ is explicitly known if, in addition, $N \equiv 1 \pmod 4$. This is discussed further below.

As a corollary, it is an easy exercise now to compute explicitly the decomposition

$$H^1\big(X(N), k\big) = H^0\big(X(N), \Omega^1\big) \oplus \overline{H^0\big(X(N), \Omega^1\big)} = L(K) \oplus \overline{L(K)}$$

into irreducible G-modules, where K is a canonical divisor. This was discussed in [KP] (over $k = \mathbb{C}$) and [Sc] (over the finite field $k = GF(N)$). Indeed, Schoen observes that the multiplicities of the irreducible representations occurring in $H^1(X(N), k)$ can be interpreted in terms of the dimension of cusp forms and number of cusps on $X(N)$.

In Sects. 6.2.1 and 6.4, applications to AG codes associated to this curve are considered (SAGE [S] was used to do some of these computations). In Sect. 6.7.1, we look at the examples $N = 7, 11$, using [GAP] to do many of the computations.

Notation Throughout this chapter, we assume that $N > 5$ is a prime, $GF(N)$ is the finite field with N elements, and $G = PSL(2, N)$.

6.2 Introduction to Algebraic Geometric Codes

Let $\mathbb{F} = GF(q)$ denote a finite field, and let $F = \overline{\mathbb{F}}$ denote the algebraic closure of $GF(q)$.

In the early 1980s, a Russian mathematician Goppa discovered a way to associated to each "nice" algebraic curve defined over a finite field a family of error-correcting codes whose length, dimension, and minimum distance can either be determined precisely or estimated in terms of some geometric parameters of the curve you started with. In this section, rather than going into detail about Goppa's general construction, we shall focus on a very special case where these constructions can be made very explicitly.

6.2.1 The Codes

If D is any divisor on X, then the Riemann–Roch space $L(D)$ is a finite-dimensional F-vector space given by

$$L(D) = L_X(D) = \left\{ f \in F(X)^\times \mid \operatorname{div}(f) + D \geq 0 \right\} \cup \{0\}, \tag{6.2.1}$$

where $\operatorname{div}(f)$ denotes the divisor of the function $f \in F(X)$. These are the rational functions whose zeros and poles are "no worse than those specified by D." Let $\ell(D)$ denote its dimension.

Let

$$D \in \operatorname{Div}(X)$$

be a divisor in $X(F)$ stabilized by G whose support is contained in $X(\mathbb{F})$. Let $P_1, \ldots, P_n \in X(\mathbb{F})$ be distinct points, and

$$E = P_1 + \cdots + P_n \in \operatorname{Div}(X)$$

be stabilized by G. This implies that G acts on the set $\operatorname{supp}(E)$ by permutation. Assume that $\operatorname{supp}(D) \cap \operatorname{supp}(E) = \emptyset$. Choose an $\mathbb{F}$-rational basis for $L(D)$ and let $L(D)_{\mathbb{F}}$ denote the corresponding vector space over $\mathbb{F}$. Let C denote the *algebraic-geometric (AG) code*

$$C = C(D, E) = \left\{ \big(f(P_1), \ldots, f(P_n)\big) \mid f \in L(D)_{\mathbb{F}} \right\}. \tag{6.2.2}$$

This is the image of $L(D)_{\mathbb{F}}$ under the evaluation map

$$\begin{aligned} &\operatorname{eval}_E : L(D) \to F^n, \\ &f \longmapsto \big(f(P_1), \ldots, f(P_n)\big). \end{aligned} \tag{6.2.3}$$

The group G acts on C by $g \in G$ sending

$$c = \big(f(P_1), \ldots, f(P_n)\big) \in C \longmapsto c' = \big(f\big(g^{-1}(P_1)\big), \ldots, f\big(g^{-1}(P_n)\big)\big),$$

where $f \in L(D)$. First, we observe that this map, denoted $\phi(g)$, is well defined. In other words, if eval_E is not injective and c is also represented by $f' \in L(D)$, so $c = (f'(P_1), \ldots, f'(P_n)) \in C$, then we can easily verify $(f(g^{-1}(P_1)), \ldots, f(g^{-1}(P_n))) = (f'(g^{-1}(P_1)), \ldots, f'(g^{-1}(P_n)))$. (Indeed, G acts on the set $\operatorname{supp}(E)$ by permutation.) This map $\phi(g)$ induces a homomorphism of G into the permutation automorphism group of the code $\operatorname{Aut}(C)$, denoted

$$\phi : G \to \operatorname{Aut}(C). \tag{6.2.4}$$

For properties of this map, see [JKT]. In particular, the following is known.

Lemma 177 *If D and E satisfy $\deg(D) > 2g$ and $\deg(E) > 2g + 2$, then ϕ and eval_E are injective.*

Proof We say that the space $L(D)$ *separates points* if for all points $P, Q \in X$, $f(P) = f(Q)$ (for all $f \in L(D)$) implies $P = Q$ (see [Ha], Chap. II, Sect. 7). By Proposition IV.3.1 in Hartshorne [Ha], D very ample implies that $L(D)$ separates points. In general, if $L(D)$ separates points, then

$$\mathrm{Ker}(\phi) = \{g \in G \mid g(P_i) = P_i,\ 1 \le i \le n\}.$$

It is known (proof of Proposition VII3.3, [Sti]) that if $n = \deg(E) > 2g + 2$, then $\{g \in G \mid g(P_i) = P_i,\ 1 \le i \le n\}$ is trivial. Therefore, if $n > 2g + 2$ and $L(D)$ separates points, then ϕ is injective. Since (see Corollary IV.3.2 in Hartshorne [Ha]) $\deg(D) > 2g$ implies that D is very ample, the lemma follows. □

Let P be the permutation automorphism group of the code $C = C(D, E)$ defined in (6.2.2). In many cases it is known that the map $\phi : G \to P$ is an isomorphism (see, for example, [JKT]). In any case, using (6.2.4), we regard C as a G-module. In particular, the (bijective) evaluation map $\mathrm{eval}_E : L(D) \to C$ in (6.2.3) is G-equivariant.

A code associated to the projective line in the manner above is referred to as a *generalized Reed–Solomon code*. The automorphism groups of such codes have been completely determined (see, for example, [JKT]).

6.2.2 The Projective Line

By way of introduction, we start with the example of the simplest curve, the projective line.

What exactly is the projective line $\mathbb{P}^1$? The analogy to keep in mind is that $\mathbb{P}^1$ is analogous to the complex plane compactified by adding the point at infinity, i.e., the Riemann sphere $\hat{\mathbb{C}}$.

Algebraically, in a rigorous treatment points are replaced by places, "valuations" on the function field $F(\mathbb{P}^1)$, which correspond to localizations of a coordinate ring $F[\mathbb{P}^1]$ (see Moreno [Mo], Sect. 1.1).

We shall, for reasons of space, emphasize intuition over precision. What is a point? $\mathbb{P}^1$ (as a set) may be thought of as the set of lines through the origin in affine space F^2. We say that two points in $F^2 - \{(0, 0)\}$ are "equivalent" if they lie on the same line (this is an equivalence relation). If $y \neq 0$, then we denote the equivalence class of (x, y) by $[a : 1]$, where $a = x/y$. If $y = 0$, then we denote the equivalence class of (x, y) by $[1 : 0]$. This notation is called the *projective coordinate* notation for elements of $\mathbb{P}^1$.

The group $GL(2, \mathbb{C})$ acts on the Riemann sphere by linear fractional ("Möbius") transformations, $z \longmapsto \frac{az+b}{cz+d}$, $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in GL(2, \mathbb{C})$. This action factors through $PGL(2, \mathbb{C})$ since scalar matrices act trivially. Similarly, $PGL(2, F)$ acts on $\mathbb{P}^1$. In fact, $\mathrm{Aut}(\mathbb{P}^1) = PGL(2, F)$.

Riemann–Roch Spaces

The only meromorphic functions on the Riemann sphere are the rational functions, so we focus on the F-valued rational functions on the $\mathbb{P}^1$, denoted $F(\mathbb{P}^1)$. Let $f \in F(\mathbb{P}^1)$, so $f(x) = \frac{p(x)}{q(x)}$ is a rational function, where x is a "local coordinate" on $\mathbb{P}^1$, and $p(x), q(x)$ are polynomials. In other notation,

$$F\big(\mathbb{P}^1\big) = F(x).$$

For example, a polynomial $f(x)$ of degree n in x is an element of $F(\mathbb{P}^1)$ which has n zeros (by the fundamental theorem of algebra) and a pole of order n at "the point at infinity", denoted ∞. (What this really means is that $f(1/x)$ has a pole of order n at $x = 0$.)

A *divisor on* $\mathbb{P}^1$ is simply a formal linear combination of points with integer coefficients, only finitely many of which are nonzero. The *divisor of* f is the formal sum of zeros of f minus the poles, counted according to multiplicity. These sums include any zero or pole at the "point at infinity" on $\mathbb{P}^1$. For any given divisor D, the set of points occurring in the formal sum defining D whose integer coefficient is nonzero is called the *support* of D, written $\mathrm{supp}(D)$. The divisor of a rational function f is denoted $\mathrm{div}(f)$. If f is, for example, a polynomial of degree n in x, then $\mathrm{div}(f) = P_1 + \cdots + P_n - n\infty$ and $\mathrm{supp}(\mathrm{div}(f)) = \{P_1, \ldots, P_n, \infty\}$, where the P_i's denote the zeros of f. Since divisors are merely formal integral combinations of points, the sum and difference of any two divisors are other divisors. The Abelian group of all divisors is denoted $\mathrm{Div}(\mathbb{P}^1)$.

Let $X = \mathbb{P}^1$, and let $F(X)$ denote the function field of X (the field of rational functions on X).

Let $\infty = [1 : 0] \in X$ denote the point at infinity. In this case, the Riemann–Roch theorem becomes

$$\ell(D) - \ell(-2\infty - D) = \deg(D) + 1.$$

It is known (and easy to show) that if $\deg(D) < 0$, then $\ell(D) = 0$, and if $\deg(D) \geq 0$, then $\ell(D) = \deg(D) + 1$.

The Action of G on $L(D)$

Let F be algebraically closed, and let $X = \mathbb{P}^1/F$, by which we mean $\mathbb{P}^1$ with base field F, so the field of rational functions is $F(\mathbb{P}^1)$. In this case, $\mathrm{Aut}(X) = PGL(2, F)$.

The action ρ of $\mathrm{Aut}(X)$ on $F(X)$ is defined by

$$\begin{aligned} \rho : \mathrm{Aut}(X) &\longrightarrow \mathrm{Aut}\big(F(X)\big), \\ g &\longmapsto \big(f \longmapsto f^g\big) \end{aligned}$$

where $f^g(x) = (\rho(g)(f))(x) = f(g^{-1}(x))$.

Note that $Y = X/G$ is also smooth and $F(X)^G = F(Y)$.

Of course, $\mathrm{Aut}(X)$ also acts on the group $\mathrm{Div}(X)$ of divisors of X, denoted $g(\sum_P d_P P) = \sum_P d_P g(P)$, for $g \in \mathrm{Aut}(X)$, P a prime divisor, and $d_P \in \mathbb{Z}$. It is easy to show that $\mathrm{div}(f^g) = g(\mathrm{div}(f))$. Because of this, if $\mathrm{div}(f) + D \geq 0$, then $\mathrm{div}(f^g) + g(D) \geq 0$ for all $g \in \mathrm{Aut}(X)$. In particular, if the action of $G \subset \mathrm{Aut}(X)$ on X leaves $D \in \mathrm{Div}(X)$ stable, then G also acts on $L(D)$. We denote this action by

$$\rho : G \to \mathrm{Aut}\big(L(D)\big).$$

A basis for the Riemann–Roch space is explicitly known for $\mathbb{P}^1$. For notational simplicity, let

$$m_P(x) = \begin{cases} x, & P = [1:0] = \infty, \\ (x-p)^{-1}, & P = [p:1]. \end{cases}$$

Lemma 178 *Let $P_0 = \infty = [1:0] \in X$ denote the point corresponding to the localization $F[x]_{(1/x)}$. For $1 \leq i \leq s$, let $P_i = [p_i : 1]$ denote the point corresponding to the localization $F[x]_{(x-p_i)}$ for $p_i \in F$. Let $D = \sum_{i=0}^{s} a_i P_i$ be a divisor, $a_k \in \mathbb{Z}$ for $0 \leq k \leq s$.*

(a) *If D is effective, then*

$$\big\{1, m_{P_i}(x)^k \mid 1 \leq k \leq a_i, 0 \leq i \leq s\big\}$$

is a basis for $L(D)$.

(b) *If D is not effective but $\deg(D) \geq 0$, then write $D = dP + D'$, where $\deg(D') = 0$, $d > 0$, and P is any point. Let $q(x) \in L(D')$ (which is a one-dimensional vector space) be any nonzero element. Then*

$$\big\{m_P(x)^i q(x) \mid 0 \leq i \leq d\big\}$$

is a basis for $L(D)$.

(c) *If $\deg(D) < 0$, then $L(D) = \{0\}$.*

The first part is Lemma 2.4 in [Lo]. The other parts follow from the definitions and the Riemann–Roch theorem.

6.3 Introduction to Modular Curves

Suppose that V is a smooth projective variety over a finite field $\mathbb{F}$. An important problem in arithmetical algebraic geometry is the calculation of the number of $\mathbb{F}$-rational points of V, $|V(\mathbb{F})|$. The work of Goppa [G1] and others have shown its importance in geometric coding theory as well.[1] We refer to this problem as the

[1] The expository paper [JS] discussed this in more detail from the computational perspective.

counting problem. In most cases it is very hard to find an explicit formula for the number of points of a variety over a finite field.

When the V arises by "reduction mod p" from a "Shimura variety" defined by certain group theoretical conditions (see Sect. 6.3.1 below), methods from non-Abelian harmonic analysis on groups can be used to find an explicit solution for the counting problem. The Arthur–Selberg trace formula [Shok] provides such a method. Using the Arthur–Selberg trace formula, an explicit formula for the counting problem has been found for Shimura varieties, thanks primarily to the work of Langlands and Kottwitz [Lan1, K1, K2].[2] The trace formula allows one (with sufficient skill and expertise!) to relate, when V is a Shimura variety, the geometric numbers $|V(k)|$ to orbital integrals from harmonic analysis ([Lab], for example), or to a linear combination of coefficients of automorphic forms ([Gel], for example), or even to representation-theoretic data ([Cas2], for example).

However, another type of application of the trace formula is very useful to the coding theorist. Moreno [Mo] first applied the trace formula in the context of Goppa codes to obtaining a new proof of a famous result of M. Tsfasman, S. Vladut, T. Zink, and Y. Ihara. (Actually, Moreno used a formula for the trace of the Hecke operators acting on the space of modular forms of weight 2, but this can be proven as a consequence of the Arthur–Selberg trace formula [DL], Sect. II.6.) This will be discussed below. We are going to restrict our attention in this chapter to the interplay between Goppa codes of modular curves, the counting problem, and the action of the automorphism group on these codes. We will give some examples using SAGE. In coding theory, curves with many rational points over finite fields are being used for construction of codes with some good specific characteristics. We discuss AG (or Goppa) codes arising from curves, first from an abstract general perspective, then turning to concrete examples associated to modular curves. We will try to explain these extremely technical ideas using a special case at the level of a typical graduate student with some background in modular forms, number theory, group theory, and algebraic geometry. For an approach similar in spirit, though from a more classical perspective, see the book of Moreno [Mo].

6.3.1 Shimura Curves

In this section we study arithmetic subgroups, arithmetical quotients, and their rational compactifications. Ihara first introduced Shimura curves, a rational compactification of $\Gamma\backslash\mathbb{H}$ where Γ is a particular discrete subgroup acting on the upper half-plane[3] $\mathbb{H}$, from a classical perspective. We shall recall them from both the classical and group-theoretical point of view. The latter perspective generalizes to higher-dimensional Shimura varieties [Del].

[2]For some introductions to this highly technical work of Langlands and Kottwitz, the reader is referred to Labesse [Lab], Clozel [Cl], and Casselman [Cas2].

[3]The space $\mathbb{H} = \{z \in \mathbb{C} \mid \mathrm{Im}\,(z) > 0\}$ is also called *the Poincaré* upper half plane.

Arithmetic Subgroups

We assume that $G = SL(2)$ is the group of 2×2 matrices with entries from an algebraically closed field Ω. In particular, the group of R-points of $SL(2)$ for a subring $R \subseteq \Omega$ with unit element 1 is defined by

$$SL(2, R) = \{g \in M(2, R) \mid \det(g) = 1\},$$

where $M(2, R)$ is the space of 2×2 matrices with entries from R. We now define congruence subgroups in $SL(2, \mathbb{Z})$. Let $SL(2, \mathbb{Z})$ be the subgroup of $SL(2, \mathbb{R})$ with integral matrices. Consider a natural number N, and let

$$\Gamma(N) = \left\{ \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in SL(2, \mathbb{Z}) \,\middle|\, \begin{matrix} a, d \equiv 1 \pmod{N} \\ b, c \equiv 0 \pmod{N} \end{matrix} \right\}.$$

We note that the subgroup $\Gamma(N)$ is a discrete subgroup of $SL(2, \mathbb{R})$, which is called the *principal congruence subgroup of level* N. Any subgroup of $SL(2, \mathbb{Z})$ that contains the principal congruence subgroup is called a *congruence subgroup*.

In general, an *arithmetic subgroup* of $SL(2, \mathbb{R})$ is any discrete subgroup Γ that is commensurable with $SL(2, \mathbb{Z})$, where *commensurability* means that the intersection $\Gamma \cap SL(2, \mathbb{Z})$ is of finite index in both Γ and $SL(2, \mathbb{Z})$. The group $\Gamma(N)$ has the property of being commensurable with $SL(2, \mathbb{Z})$.

Riemann Surfaces as Algebraic Curves

Note that the group $SL(2, \mathbb{R})$ acts on $\mathbb{H}$ by

$$g \cdot z = (az + b)(cz + d)^{-1} = \frac{az + b}{cz + d},$$

where $z \in \mathbb{H}$, $g = \left[\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right] \in SL(2, \mathbb{R})$.

We emphasize that the action of $SL(2, \mathbb{R})$ on $\mathbb{H}$ is *transitive*, i.e., for any two points $w_1, w_2 \in \mathbb{H}$, there is an element $g \in SL(2, \mathbb{R})$ such that $w_2 = g \cdot w_1$. This can easily be proved. We also emphasize that there are subgroups of $SL(2, \mathbb{R})$ for which the action is not transitive; among them, the class of arithmetic subgroups are to be mentioned. For example, the group $SL(2, \mathbb{Z})$ does not act transitively on $\mathbb{H}$, and the set of orbits of the action of $SL(2, \mathbb{Z})$ on $\mathbb{H}$ (and similarly any arithmetic subgroup) is infinite. We call the *arithmetic quotient* $\Gamma \backslash \mathbb{H}$ the set of orbits of the action of an arithmetic subgroup Γ on $\mathbb{H}$.

Example 179 Take Γ to be the *Hecke subgroup* $\Gamma_0(N)$ defined by

$$\Gamma_0(N) = \left\{ \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in SL(2, \mathbb{Z}) \,\middle|\, c \equiv 0 \pmod{N} \right\}$$

for a natural number N. This is a congruence subgroup, and $Y_0(N) = \Gamma_0(N)\backslash\mathbb{H}$ is an arithmetic quotient. Such a quotient is not a compact subset, nor a bounded one; it is however a subset with finite measure (volume) under the non-Euclidean measure induced on the quotient from the group $SL(2,\mathbb{R})$ which is a locally compact group and induces the invariant volume element $\frac{dx\wedge dy}{y^2}$, where x, y are the real and complex parts of an element $z \in \mathbb{H}$.

We now recall the basic ideas that turn an arithmetic quotient of the form $\Gamma\backslash\mathbb{H}$ into an algebraic curve. Let $\Gamma \subset SL(2,\mathbb{Q})$ be an arithmetic subgroup. The topological boundary of $\mathbb{H}$ is $\mathbb{R}$ and a point ∞. For the rational compactification of $\mathbb{H}$, we do not need to consider all the boundaries $\mathbb{R}$ and $\{\infty\}$. In fact, we need only to add to $\mathbb{H}$ the cusps of Γ (a *cusp* of Γ is an element of $\mathbb{Q}$ that is fixed under the action of an element $\gamma \in \Gamma$ with the property that $|\operatorname{tr}(\gamma)| = 2$). Any two cusps x_1, x_2 such that $\delta \cdot x_2 = x_1$ for an element $\delta \in \Gamma$ are called *equivalent*. Let $C(\Gamma)$ be the set of inequivalent cusps of Γ. Then $C(\Gamma)$ is finite. We add this set to $\mathbb{H}$ and form the space $\mathbb{H}^* = \mathbb{H} \cup C(\Gamma)$. This space will be equipped with certain topology such that a basis of the neighborhoods of the points of $\mathbb{H}^*$ is given by three types of open sets; if a point in $\mathbb{H}^*$ is lying in $\mathbb{H}$, then its neighborhoods consist of the usual open discs in $\mathbb{H}$; if the point is ∞, i.e., the cusp ∞, then its neighborhoods are the set of all points lying above the line $\operatorname{Im}(z) > \alpha$ for any real number α; if the point is a cusp different from ∞ which is a rational number, then the system of neighborhoods of this point are the union of the cusp and the interior of a circle in $\mathbb{H}$ tangent to the cusp. Under the topology whose system of open neighborhoods we just explained, $\mathbb{H}^*$ becomes a Hausdorff nonlocally compact space. The quotient space $\Gamma\backslash\mathbb{H}^*$ with the quotient topology is a compact Hausdorff space. We refer to this compact quotient as the *rational compactification* of $\Gamma\backslash\mathbb{H}$. For a detailed discussion, we refer the reader to [Shim].

When the arithmetic group is a congruence subgroup of $SL(2,\mathbb{Z})$, the resulting algebraic curve is called a *modular curve*. For example, the rational compactification of $Y(N) = \Gamma(N)\backslash\mathbb{H}$ is denoted by $X(N)$, and the compactification of $Y_0(N) = \Gamma_0(N)\backslash\mathbb{H}$ by $X_0(N)$.

Example 180 Let $N = 1$. Then $\Gamma = \Gamma(1) = SL(2,\mathbb{Z})$. In this case, $C(\Gamma) = \{\infty\}$, since all rational cusps are equivalent to the cusp ∞. So $\mathbb{H}^* = \mathbb{H} \cup \{\infty\}$, and $\Gamma\backslash\mathbb{H}^*$ will be identified by $\Gamma\backslash\mathbb{H} \cup \{\infty\}$. This may be seen as adding ∞ to the fundamental domain $\mathcal{F}_1 = \mathcal{F}$ of $SL(2,\mathbb{Z})$ that consists of all complex numbers $z \in \mathbb{H}$ in the upper half-plane with $|z| \geq 1$ and $|\operatorname{Re}(z)| \leq \frac{1}{2}$.

The rational compactification of $\Gamma\backslash\mathbb{H}$ turns the space $\Gamma\backslash\mathbb{H}^*$ into a compact Riemann surface (cf. [Shim]) and so into an algebraic curve (cf. [Nara] or [SS]).

In general, it is easiest to work with those arithmetic subgroups which are torsion free, and we shall assume from this point on that the arithmetic subgroups we deal with have this property. For example, $\Gamma(N)$ and $\Gamma_0(N)$ for $N \geq 3$ are torsion free.

An Adelic View of Arithmetic Quotients

Consider the number field $\mathbb{Q}$, the field of rational numbers. Let $\mathbb{Q}_p$ be the (*p-adic*) completion of $\mathbb{Q}$ under the p-adic absolute value $|\cdot|_p$, where $|a/b|_p = p^{-n}$ whenever a, b are integers and $a/b = p^n \prod_{\ell \neq p \text{ prime}} \ell^{e_\ell}$, $n, e_\ell \in \mathbb{Z}$. (Roughly speaking, $\mathbb{Q}_p$ is a set of Laurent series in p whose coefficients belong to $\mathbb{Z}/p\mathbb{Z}$.) Under the ordinary absolute value, the completion of $\mathbb{Q}$ is $\mathbb{R}$, also denoted $\mathbb{Q}_\infty = \mathbb{R}$. These are topological fields (under the metric topology) and the ring of integers of $\mathbb{Q}_p$,

$$\mathbb{Z}_p = \{x \in \mathbb{Q}_p \,|\, |x| \leq 1\},$$

is a maximal compact open subring of $\mathbb{Q}_p$. The ring of *adeles* of $\mathbb{Q}$ is the commutative ring $\mathbb{A}$ that is given by the restricted direct product

$$\mathbb{A} = \left\{ (x_\infty, x_2, \dots) \in \mathbb{R} \times \prod_p \mathbb{Q}_p \;\middle|\; \text{all but a finite number of } x_p \in \mathbb{Z}_p \right\}.$$

In the product topology, $\mathbb{A}$ is a locally compact ring. If $\mathbb{A}_f$ denotes the set of adeles omitting the $\mathbb{R}$-component x_∞, then $\mathbb{A}_f$ is called the *ring of finite adeles*, and we can write $\mathbb{A} = \mathbb{R} \times \mathbb{A}_f$. Under the diagonal embedding, $\mathbb{Q}$ is a discrete subgroup of $\mathbb{A}$.

We now consider the group $G = GL(2)$. For a choice of an open compact subgroup $K_f \subset G(\mathbb{A}_f)$, it is known that we can write the arithmetic quotient (which was originally attached to an arithmetic subgroup of $\Gamma \subset SL(2, \mathbb{Q})$) as the following quotient:

$$Y(K_f) = G(\mathbb{Q}) \backslash \left[\mathbb{H} \times \left(G(\mathbb{A}_f)/K_f \right) \right] = \Gamma \backslash H, \tag{6.3.1}$$

where

$$\Gamma = G(\mathbb{Q}) \cap G(\mathbb{R}) K_f. \tag{6.3.2}$$

Thus our arithmetic subgroup Γ is completely determined by K_f. From now on we assume that K_f has been chosen so that Γ is torsion free.

Definition 181 Let $G = GL(2)$. To G is associated the Shimura variety $Sh(G)$ as follows. Let $N \geq 3$ be a natural number. Let $\Gamma(N)$ be the congruence subgroup of level N of $SL(2, \mathbb{Z})$, and $K = SO(2, \mathbb{R})$ the orthogonal group of 2×2 real matrices A with determinant 1 satisfying ${}^t AA = I_2$, where I_2 denotes the 2×2 identity matrix. Then

$$Y(N) = \Gamma(N) \backslash \mathbb{H} \cong \Gamma(N) \backslash G(\mathbb{R}) / K.$$

We call this the *modular space of level N*. Let

$$K_f(N) = \left\{ g \in G\left(\prod_p \mathbb{Z}_p \right) \;\middle|\; g \equiv I_2 \pmod{N} \right\}$$

be the *open compact subgroup of* $G(\mathbb{A}_f)$ *of level* N. Then the modular space of level N can be written as

$$Y(N) \cong G(\mathbb{Q})\backslash G(\mathbb{A})/KK_f(N) = G(\mathbb{Q})\backslash\big[\mathbb{H} \times \big(G(\mathbb{A}_f)/K_f(N)\big)\big].$$

Thus,

$$X\big(K_f(N)\big) \cong Y(N).$$

Taking the projective limit over $K_f(N)$ by letting N get large (which means that $K_f(N)$ gets small), we see that $\lim_N Y(N) = G(\mathbb{Q})\backslash[\mathbb{H} \times G(\mathbb{A}_f)]$. Then the (complex points of the) *Shimura curve* $Sh(G)$ associated to $G = SL(2)$ is defined by

$$Sh(G)(\mathbb{C}) = G(\mathbb{Q})\backslash\big[\mathbb{H} \times G(\mathbb{A}_f)\big]. \tag{6.3.3}$$

Many mathematicians have addressed the natural questions below.

- What field are the curves $X(N)$, $X_0(N)$ "naturally" defined over?
- How can they be described explicitly using algebraic equations?

Regarding the first question, from the general theory of Shimura varieties we know that for each reductive group G defined over $\mathbb{Q}$ satisfying the axioms of Sect. 2.1.1 in [Del], there is an algebraic number field $E = E_G$ over which a Shimura variety $Sh(G)$ is defined [Del]. In fact, the Shimura curves $X(N)$ and $X_0(N)$ are regular schemes proper over $\mathbb{Z}[1/N]$ (more precisely, over $\mathrm{Spec}(\mathbb{Z}[1/N])$).[4]

Regarding the second question, it is possible to find a *modular polynomial* $H_N(x, y)$ of degree

$$\mu(N) = N\prod_{p|N}\left(1 + \frac{1}{p}\right) \tag{6.3.4}$$

for which $H_N(x, y) = 0$ describes (an affine patch of) $X_0(N)$. Let

$$G_k(q) = 2\zeta(k) + 2\frac{(2i\pi)^k}{(k-1)!}\sum_{n=1}^{\infty}\sigma_{k-1}(n)q^n,$$

where $q = e^{2\pi iz}$, $z \in \mathbb{H}$, $\sigma_r(n) = \sum_{d|n} d^r$, and let

$$\Delta(q) = 60^3 G_4(q)^3 - 27 \cdot 140^2 G_6(q)^2 = q\prod_{n=1}^{\infty}\big(1 - q^n\big)^{24}.$$

[4]This result was essentially first proved by Igusa [Ig] (from the classical perspective). See also [TV], Theorem 4.1.48, and [Cas1] for an interesting discussion of what happens at the "bad primes," and Deligne's paper in the same volume as [Cas1].

Define the *j-invariant* by

$$j(q) = 1728 \cdot 60^3 G_4(q)^3/\Delta(q)$$
$$= q^{-1} + 744 + 196884q + 21493760q^2 + 864299970q^3 + \cdots .$$

(More details on Δ and j can be found, for example, in [Shim] or [Kob].) The key property satisfied by H_N is $H_N(j(q), j(q^N)) = 0$. It is interesting to note in passing that when N is such that the genus of $X_0(N)$ equals 0 (i.e., $N \in \{1, 3, 4, 5, 6, 7, 8, 9, 12, 13, 16, 18, 25\}$ [Kn]), then this implies that $(x, y) = (j(q), j(q^N))$ parameterizes $X_0(N)$. In general, comparing q-coefficients allows one to compute H_N for relatively small values of N. (The SAGE command `ClassicalModularPolynomialDatabase()` loads a database which allows one to compute this expression.) However, even for $N = 11$, some of the coefficients can involve one hundred digits or more. The cases $N = 2, 3$ are given, for example, in Elkies [E1].

Example 182 The following SAGE commands which illustrate this require David Kohel's database `database_kohel` be loaded first.[5]

SAGE

```
sage: C = ClassicalModularPolynomialDatabase()
sage: f = C[3]
sage: f
-j0^3*j1^3 + 2232*j0^3*j1^2 + 2232*j0^2*j1^3 + j0^4 \
- 1069956*j0^3*j1 + 2587918086*j0^2*j1^2 - 1069956*j0*j1^3\
+ j1^4 + 36864000*j0^3 + 8900222976000*j0^2*j1 + 8900222976000*j0*j1^2\
+ 36864000*j1^3 + 452984832000000*j0^2 - 770845966336000000*j0*j1\
+ 452984832000000*j1^2 + 1855425871872000000000*j0 + 1855425871872000000000*j1
```

This is basically (20) in [E1].

The paper by Cohen [Co] determines the asymptotic size of the largest coefficient of H_N (normalized to have leading coefficient equal to 1). She shows that the largest coefficient grows like $N^{c\mu(N)}$, where $c > 0$ is a constant, and μ is as in (6.3.4). More practical equations for (some of) the $X_0(N)$ are given in Hibino and Murabayashi [HM], Shimura [ShimM], Rovira [Ro], Frey and Müller [FM], Birch [B], and Table 6.1 in Sect. 6.5 below.

For deeper study of Shimura varieties and the theory of canonical models, we refer the reader to [Del, Lan2], and [Shim].

[5]Type `optional_packages()` for the name of the latest version of this database. This loads both `ClassicalModularPolynomialDatabase` and `AtkinModularPolynomialDatabase`.

6.3.2 Hecke Operators and Arithmetic on $X_0(N)$

In this section we recall some well-known though relatively deep results on $X_0(N)(GF(p))$, where p is a prime not dividing N. These shall be used in the discussion of the Tsfasman, Vladut, Zink, and Ihara result later.

First, some notation: let $S_2(\Gamma_0(N))$ denote the space of holomorphic cusp forms of weight 2 on $\Gamma_0(N)\backslash H$. Let $T_p : S_2(\Gamma_0(N)) \to S_2(\Gamma_0(N))$ denote the *Hecke operator* defined by

$$T_p f(z) = f(pz) + \sum_{i=0}^{p-1} f\left(\frac{z+i}{p}\right), \quad z \in H.$$

Define T_{p^k} inductively by

$$T_{p^k} = T_{p^{k-1}} T_p - pT_{p^{k-2}}, \quad T_1 = 1,$$

and define the modified Hecke operators U_{p^k} by

$$U_{p^k} = T_{p^k} - pT_{p^{k-2}}, \quad U_p = T_p,$$

for $k \geq 2$. The Hecke operators may be extended to the positive integers by demanding that they be multiplicative.

Theorem 183 ("Congruence relation" of Eichler–Shimura [Mo], Sect. 5.6.7, or [St1]) *Let $q = p^k$, $k > 0$ an integer. If p is a prime not dividing N, then*

$$\mathrm{Tr}(T_p) = p + 1 - \left|X_0(N)\big(GF(p)\big)\right|.$$

More generally,

$$\mathrm{Tr}(T_q - pT_{q/p^2}) = q + 1 - \left|X_0(N)\big(GF(q)\big)\right|.$$

Example 184 One may try to compute the trace of the Hecke operators T_p acting on the space of holomorphic cusp forms of weight 2, $S_2(\Gamma_0(N))$, by using either the Eichler–Shimura congruence relation, which we give below (see Theorem 183), or by using some easier but ad hoc ideas going back to Hecke which work in special cases. One simple idea is noting that $S_2(\Gamma_0(N))$ is spanned by simultaneous eigenforms of the Hecke operators (see, for example, Proposition 51 in Chap. III of [Kob]). In this case, it is known that the Fourier coefficient a_p, p prime not dividing N, of a normalized (to have leading coefficient $a_1 = 1$) eigenform is the eigenvalue of T_p (see, for example, Proposition 40 in Chap. III of [Kob]). If $S_2(\Gamma_0(N))$ is one-dimensional, then any element in that space $f(z)$ is such an eigenform.

The modular curve $X_0(11)$ is of genus 1, so there is (up to a nonzero constant factor) only one holomorphic cusp form of weight 2 in $S_2(\Gamma_0(11))$ (see Theorem 186 below). There is a well-known construction of this form (see [O2] or [Gel], Example 5.1), which we recall below. As we noted above, the pth coefficient a_p (p a prime distinct from 11) of its Fourier expansion is known to satisfy $a_p = \mathrm{Tr}(T_p)$. These will be computed using SAGE.

Let $q = e^{2\pi i z}$, $z \in \mathbb{H}$, and consider *Dedekind's η-function*,

$$\eta(z) = e^{2\pi i z/24} \prod_{n=1}^{\infty} (1 - q^n).$$

Then

$$f(z) = \eta(z)^2 \eta(11z)^2 q \prod_{n=1}^{\infty} (1 - q^n)^2 (1 - q^{11n})^2$$

is an element[6] of $S_2(\Gamma_0(11))$. One can compute the q-expansion of this form using SAGE's `ModularForms(Gamma0(11),2)` command:

$$f(z) = q - 2q^2 - q^3 + 2q^4 + q^5 + 2q^6 - 2q^7 \cdots.$$

SAGE

```
sage: M = ModularForms(Gamma0(11),2)
sage: M.q_expansion_basis(prec = 8)

[
q - 2*q^2 - q^3 + 2*q^4 + q^5 + 2*q^6 - 2*q^7 + O(q^8),
1 + 12/5*q + 36/5*q^2 + 48/5*q^3 + 84/5*q^4 + 72/5*q^5
    + 144/5*q^6 + 96/5*q^7 + O(q^8)
]
```

For example, the above expansion tells us that $\mathrm{Tr}(T_3) = \mathrm{Tr}(U_3) = -1$. The curve $X_0(11)$ is of genus 1 and is isogenous to the elliptic curve C with Weierstrass model $y^2 + y = x^3 - x^2$. Over the field with $p = 3$ elements, there are $|X_0(11)(GF(3))| = p + 1 - \mathrm{Tr}(T_p) = 5$ points in $C(GF(3))$, including ∞:

$$C\big(GF(3)\big) = \big\{[0,0], [0,2], [1,0], [1,2], \infty\big\}.$$

For this, one uses the SAGE commands

SAGE

```
sage: F = GF(3)
sage: P.<x> = PolynomialRing(F,"x")
sage: f = x^3-x^2; h = 1
sage: C = HyperellipticCurve(f, h)
sage: C.rational_points()
[(0 : 0 : 1), (0 : 1 : 0), (0 : 2 : 1), (1 : 0 : 1), (1 : 2 : 1)]
```

[6] In fact, if we write $f(z) = \sum_{n=1}^{\infty} a_n q^n$, then

$$\zeta_C(s) = \big(1 - p^{-s}\big)^{-1} \prod_{p \neq 11} \big(1 - a_p p^{-s} + p^{1-2s}\big)^{-1}$$

is the global Hasse–Weil zeta function of the elliptic curve C of conductor 11 with Weierstrass model $y^2 + y = x^3 - x^2$ [Gel] (p. 252).

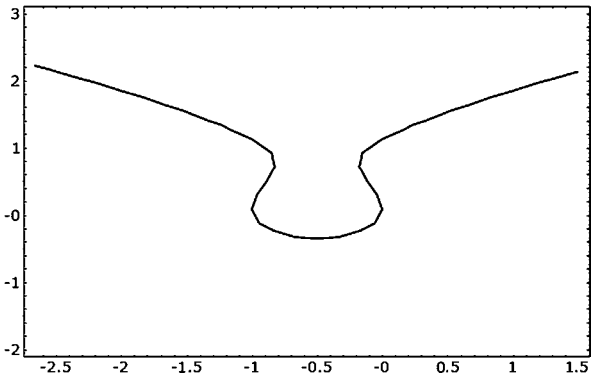


Fig. 6.1 The elliptic curve $y^2 + y = x^3 - x^2$ over $\mathbb{R}$

A plot of the real points of this elliptic curve is given in Fig. 6.1.

For a representation-theoretic discussion of this example, see [Gel], Sect. 14.

For an example of an explicit element of $S_2(\Gamma_0(32))$, see Koblitz [Kob] (Sect. 5 in Chap. II and (3.40) in Chap. III). For a remarkable theorem which illustrates how far this η-function construction can be extended, see Morris' theorem in Sect. 2.2 of [Ro].

To estimate a_{p^k}, one may appeal to an explicit expression for $\mathrm{Tr}(T_{p^k})$ known as the "Eichler–Selberg trace formula", which we discuss next.

6.3.3 Eichler–Selberg Trace Formula

In this subsection, we recall the version of the trace formula for the Hecke operators due to Duflo and Labesse [DL], Sect. 6.

Let k be an even positive integer, and let Γ be a congruence subgroup as in (6.3.2). Let S denote a complete set of representatives of $G(\mathbb{Q})$-conjugacy classes of $\mathbb{R}$-elliptic elements in Γ ($\mathbb{R}$-elliptic elements are those that are conjugate to an element of $SO(2,\mathbb{R})$, the orthogonal group). For $\gamma \in S$, let $w(\gamma)$ denote the cardinality of the centralizer of γ in Γ. If $r(\theta) = \left(\begin{smallmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{smallmatrix}\right)$, then let $\theta_\gamma \in (0, 2\pi)$ denote the element for which $\gamma = r(\theta_\gamma)$. Let τ_m denote the image in $G(\mathbb{A}_f)$ of the set of matrices in $GL(2,\mathbb{A}_f)$ having coefficients in $\hat{\mathbb{Z}} = \prod_{p<\infty} \mathbb{Z}_p$ and determinant in $m\hat{\mathbb{Z}}$. Consider the subspace $S_k(\Gamma) \subset L^2(\Gamma\backslash H)$ formed by the functions f satisfying

- $f(\gamma z) = (cz+d)^k f(z)$ for all $\gamma = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma$, $x \in H$,
- f is a holomorphic cusp form.

This is the space of holomorphic cusp forms of weight k on $\mathbb{H}$.

Let

$$\epsilon\left(\sqrt{m}\right) = \begin{cases} 1, & m \text{ is a square,} \\ 0, & \text{otherwise,} \end{cases}$$

and let

$$\delta_{i,j} = \begin{cases} 1, & i = j, \\ 0, & \text{otherwise.} \end{cases}$$

Theorem 185 (Eichler–Selberg trace formula) *Let $k > 0$ be an even integer, and $m > 0$ an integer. The trace of T_m acting on $S_k(\Gamma)$ is given by*

$$\operatorname{Tr}(T_m) = \delta_{2,k} \sum_{d|m} b + \epsilon\big(\sqrt{m}\big)\left(\frac{k-1}{12} m^{(k-2)/2} - \frac{1}{2} m^{(k-1)/2}\right) - \sum_{\gamma \in S \cap \tau_m} w(\gamma)^{-1} m^{(k-2)/2} \frac{\sin((k-1)\theta_\gamma)}{\sin(\theta_\gamma)} - \sum_{d|m,\ d^2<m} b^{k-1}.$$

Remark 24 Let $k = 2$, $m = p^2$, $\Gamma = \Gamma_0(N)$, and $N \to \infty$ in the above formula. It is possible to show that the Eichler–Selberg trace formula implies

$$\operatorname{Tr}(T_{p^2}) = g\big(X_0(N)\big) + O(1) \tag{6.3.5}$$

as $N \to \infty$. The proof of this estimate (see [Mo], Chap. 5, or [LvdG], Sect. V.4) uses the explicit formula given below for $g(X_0(N)) = \dim(S_2(\Gamma_0(N)))$, which we shall also make use of later.

Theorem 186 (Hurwitz–Zeuthen formula [Shim])[7] *The genus of $X_0(N)$ is given by*

$$g\big(X_0(N)\big) = \dim\big(S_2\big(\Gamma_0(N)\big)\big) = 1 + \frac{1}{12}\mu(N) - \frac{1}{4}\mu_2(N) - \frac{1}{3}\mu_3(N) - \mu_\infty(N),$$

where μ is as in (6.3.4),

$$\mu_2(N) = \begin{cases} \prod_{p|N \text{ prime}} \big(1 + \big(\frac{-4}{p}\big)\big), & \gcd(4, N) = 1, \\ 0, & 4|N, \end{cases}$$

$$\mu_3(N) = \begin{cases} \prod_{p|N \text{ prime}} \big(1 + \big(\frac{-3}{p}\big)\big), & \gcd(2, N) = 1 \text{ and } \gcd(9, N) \neq 9, \\ 0, & 2|N \text{ or } 9|N, \end{cases}$$

and

$$\mu_\infty(N) = \sum_{d|N} \phi\big(\gcd(d, N/d)\big),$$

where ϕ is Euler's totient function, and $(\frac{\cdot}{p})$ is Legendre's symbol.

[7] The genus formulas for $X_0(N)$ given in [Shim] and [Kn] both apparently contain a (typographical) error. The problem is in the μ_2 term, which should contain a Legendre symbol $(\frac{-4}{n})$ instead of $(\frac{-1}{n})$. See, for example, [Ei] for a correct generalization.

Estimate (6.3.5) and the Eichler–Shimura congruence relation imply

$$\begin{aligned}\left|X_0(N)\left(GF\left(p^2\right)\right)\right| &= p^2+1-\mathrm{Tr}(T_{p^2}-pI)\\ &= p^2+1-\mathrm{Tr}(T_{p^2})+p\cdot\dim\left(S_2\left(\Gamma_0(N)\right)\right)\\ &= p^2+1-\left(g\left(X_0(N)\right)+O(1)\right)+p\cdot g\left(X_0(N)\right)\\ &= (p-1)g\left(X_0(N)\right)+O(1)\end{aligned} \tag{6.3.6}$$

as $N\to\infty$.

6.3.4 Modular Curves $X(N)$

Let H denote the complex upper half-plane, let $\mathbb{H}^*=\mathbb{H}\cup\mathbb{Q}\cup\{\infty\}$, and recall that $SL(2,\mathbb{Q})$ acts on $\mathbb{H}^*$ by fractional linear transformations. Let $X(N)$ denote the modular curve defined over $\mathbb{Q}$ whose complex points are given by $\Gamma(N)\backslash\mathbb{H}^*$, where

$$\Gamma(N)=\left\{\begin{pmatrix}a & b\\ c & d\end{pmatrix}\in SL_2(\mathbb{Z})\,\middle|\,a-1\equiv d-1\equiv b\equiv c\equiv 0\ (\mathrm{mod}\ N)\right\}.$$

Throughout this paper, we will assume that N is prime and $N>6$. In this case, the genus of $X(N)$ is given by the formula

$$g=1+\frac{(N-6)(N^2-1)}{24}.$$

For example, $X(7)$ is of genus 3, and $X(11)$ is of genus 26.

Let N be a prime, and, for $j\in\mathbb{Z}/N\mathbb{Z}$, let y_j be variables satisfying

$$\begin{aligned}&y_j+y_{-j}=0,\\ &y_{a+b}y_{a-b}y_{c+d}y_{c-d}+y_{a+c}y_{a-c}y_{d+b}y_{d-b}+y_{a+d}y_{a-d}y_{b+c}y_{b-c}=0\end{aligned} \tag{6.3.7}$$

for all $a,b,c,d\in\mathbb{Z}/N\mathbb{Z}$. These are Klein's equations for $X(N)$ (see Adler [A1] or Ritzenthaler [R1]).

Example 187 When $N=7$, this reduces to

$$y_1^3y_2-y_2^3y_3-y_3^3y_1=0,$$

the famous Klein quartic.

When $N = 11$, the 20 equations which arise reduce to the 10 equations

$$\begin{aligned}
-y_1^2y_2y_3 + y_2y_4y_5^2 + y_3^2y_4y_5 &= 0,\\
-y_1^3y_4 + y_2y_4^3 - y_3^3y_5 &= 0,\\
-y_1y_3^3 - y_1^3y_5 + y_2^3y_4 &= 0,\\
-y_1^2y_3y_4 + y_1y_3y_5^2 + y_2^2y_4y_5 &= 0,\\
-y_1^2y_2y_5 + y_1y_3y_4^2 - y_2^2y_3y_5 &= 0,\\
y_1^3y_2 - y_3y_5^3 - y_4^3y_5 &= 0,\\
y_1y_5^3 - y_2^3y_3 + y_3^3y_4 &= 0,\\
-y_1y_2^2y_4 + y_1y_4^2y_5 + y_2y_3^2y_5 &= 0,\\
y_1y_2^3 + y_2y_5^3 - y_3y_4^3 &= 0,\\
y_1y_2y_3^2 + y_1y_4y_5^2 - y_2y_3y_4^2 &= 0.
\end{aligned}$$

The curve $X(N)$ over a field k parameterizes pairs of an elliptic curve over k and a subgroup of order N of the group structure on the elliptic curve. This can be extended to fields of positive characteristic if $X(N)$ has good reduction. Since Klein's equations have integer coefficients, they can also be extended to an arbitrary field k. However, Velu [V] (see also Ritzenthaler [R3]) has shown that $X(N)$ has good reduction over fields of characteristic p where p does not divide N (in our case, $p \neq N$, since N is itself assumed to be a prime).

Let

$$G = PSL_2(\mathbb{Z}/N\mathbb{Z}) \cong \overline{\Gamma(1)}/\overline{\Gamma(N)},$$

where the overline denotes the image in $PSL_2(\mathbb{Z})$. This group acts on $X(N)$. (In characteristic 0, see [Shim]; in characteristic $\ell > 0$, see [R1].) When $N > 2$ is prime, $|G| = N(N^2-1)/2$.

Definition 188 When X has good reduction to a finite field k and, in addition, the characteristic ℓ of k does not divide $|G|$, we say that ℓ is *good*.

If k is a field of good characteristic, the automorphism group of $X(N)$ is known to be $PSL(2, N)$ [BCG].

The action of $G = SL_2(\mathbb{Z}/N\mathbb{Z})$ on the set of points of the projective curve defined by Klein's equations is described in [R1] (see also [A2, R2]). The element $g = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in G$ sends $(y_j)_{j\in\mathbb{Z}/N\mathbb{Z}} \in X(N)$ to $(\rho(g)y_j)_{j\in\mathbb{Z}/N\mathbb{Z}} \in X(N)$, where

$$\rho(g)(y_j) = \sum_{t\in\mathbb{Z}/N\mathbb{Z}} \zeta^{b(aj^2+2jtc)+t^2cd} y_{aj+tc}$$

with ζ denoting a primitive Nth root of unity in k.

Remark 25 When the formulas for the special cases $\rho\left(\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right)$, $\rho\left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right)$, and $\rho\left(\begin{smallmatrix} a & 0 \\ 0 & a^{-1} \end{smallmatrix}\right)$ are written down separately, the similarity with the Weil representation for $SL_2(\mathbb{Z}/N\mathbb{Z})$ is striking (see also [A2]).

6.4 Application to Codes

In this section we discuss connections of our previous results with the theory of error-correcting codes.

Assume that ℓ is a good prime. Also, assume that k contains all the character values of G and that k is finite, where k denotes the field of definition of the reduction of X mod ℓ. (The point is that we want to be able to work over a separable algebraic closure $\overline{k}$ of k but then be able to take $\mathrm{Gal}(\overline{k}/k)$-fixed points to obtain our results.) We recall some background on AG codes following [JKT, JK1, JK2].

As an amusing application of our theory, we show how to easily recover some results of Tsfasman and Vladut on AG codes associated to modular curves.

First, we recall some notation and results from [TV]. Let $A_N = \mathbb{Z}[\zeta_N, 1/N]$, where $\zeta_N = e^{2\pi i/N}$, let K_N denote the quadratic subfield of $\mathbb{Q}(\zeta_N)$, and let $B_N = A_N \cap K_N$. There is a scheme $X(N)/\mathbb{Z}[1/N]$ which represents a moduli functor "parameterizing" elliptic curves E with a level N structure α_N. There is a scheme $X_P(N)/\mathbb{Z}[1/N]$ which represents a moduli functor "parameterizing" elliptic curves E with a "projective" level N structure β_N. If P is a prime ideal in the ring of integers $\mathcal{O}_{K_N}$ dividing ℓ, then the reduction of the form of $X(N)$ defined over K_N, denoted $X(N)/P$, is a smooth projective absolutely irreducible curve over the residue field $k(P)$, with a $PSL_2(\mathbb{Z}/N\mathbb{Z})$-action commuting with the reduction. Similarly, with $X(N)$ replaced by $X_P(N)$. Recall from Sect. 4.1.3 of [TV] that

$$k(P) = \begin{cases} GF(\ell^2), & (\ell) = P, \\ GF(\ell), & (\ell) = PP'. \end{cases}$$

Let $X'_N = X_P(N)/P$, and let

$$\psi'_N : X'_N \to X'_N/PSL_2(\mathbb{Z}/N\mathbb{Z}) \cong \mathbb{P}^1$$

denote the quotient map. Let D_∞ denote the reduced orbit of the point ∞ (in the sense of Borne), so $\deg(D_\infty) = |G|/N$. Let $D = rD_\infty$, for $r \geq 1$. According to [TV], in general, this divisor is actually defined over $GF(\ell)$, not just $k(P)$. Moreover, $\deg(D) = r \cdot (N^2 - 1)/2$. Let $E = P_1 + \cdots + P_n$ be the sum of all the supersingular points of X'_N, and let

$$C = C\left(X'_N, D, E\right) = \left\{\left(f(P_1), \ldots, f(P_n)\right) \mid f \in L(D)\right\}$$

denote the AG code associated to X'_N, D, E. This is a G-module, via (6.2.4). Moreover, choosing r suitably yields a "good" family of codes with large automorphism group.

In fact, if D is "sufficiently large" (so, D is nonspecial, and both ϕ and eval_E are injective), then the Brauer-character analogs of formulas in Joyner and Ksir [JK1] give not only the G-module structure of each $L(rD_\infty)$, but that of C as well.

See also Remark 4.1.66 in [TV].

AG Codes Associated to $X(7)$ We focus on the Klein quartic. We also use Elkies [E1, E2] as general references.

Let $\mathbb{F} = GF(43)$. This field contains 7th roots of unity ($\zeta_7 = 41$), cube roots of unity ($\zeta_3 = 36$), and the square root of -7 (take $\sqrt{-7} = 6$). Consider

$$\rho_1 = \begin{pmatrix} {\zeta_7}^4 & 0 & 0 \\ 0 & {\zeta_7}^2 & 0 \\ 0 & 0 & \zeta_7 \end{pmatrix} = \begin{pmatrix} 16 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 41 \end{pmatrix} \in M_{3\times 3}(\mathbb{F}),$$

$$\rho_2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix},$$

and

$$\rho_3 = \begin{pmatrix} (\zeta_7 - {\zeta_7}^6)/\sqrt{-7} & ({\zeta_7}^2 - {\zeta_7}^5)/\sqrt{-7} & ({\zeta_7}^4 - {\zeta_7}^3)/\sqrt{-7} \\ ({\zeta_7}^2 - {\zeta_7}^5)/\sqrt{-7} & ({\zeta_7}^4 - {\zeta_7}^3)/\sqrt{-7} & (\zeta_7 - {\zeta_7}^6)/\sqrt{-7} \\ ({\zeta_7}^4 - {\zeta_7}^3)/\sqrt{-7} & (\zeta_7 - {\zeta_7}^6)/\sqrt{-7} & ({\zeta_7}^2 - {\zeta_7}^5)/\sqrt{-7} \end{pmatrix}$$

$$= \begin{pmatrix} 11 & 37 & 39 \\ 37 & 39 & 11 \\ 39 & 11 & 37 \end{pmatrix}.$$

Elkies [E2] points out that the matrix expression for ρ_3 in terms of roots of unity can be found in Klein's 1879 paper on (what is now known as) the Klein quartic.

It may be checked that these matrices preserve the form

$$\phi(x, y, z) = x^3y + y^3z + z^3x$$

over $\mathbb{F}$. They generate the subgroup $G \cong PSL_2(7)$ of order 168 in $PGL(3, \mathbb{F})$. The Klein curve $x^3y + y^3z + z^3x = 0$, denoted here by X, has no other automorphisms in characteristic 43, so $G = \mathrm{Aut}_{\mathbb{F}}(X)$.

Let D_∞ denote the reduced orbit of the point ∞, so $\deg(D_\infty) = |G|/N = 24$, and let $D = rD_\infty$. Let $E = P_1 + \cdots + P_n$ denote the sum of the remaining $\mathbb{F}(P)$-rational points of X, so D and E have disjoint support.

If C is as in (6.2.2), then map ϕ in (6.2.4) is injective. Since eval_E is injective as well, the G-module structure of C is the same as that of $L(D)$, which is known thanks to the Brauer-character analog of the formula (6.7.1). See also Example 3 in [JK1].

The 80 points of $X(\mathbb{F})$ are

$$\{(0:1:0),(0:0:1),(1:0:0),(19:9:1),(36:9:1),(31:9:1),(19:27:1),$$
$$(1:38:1),(27:38:1),(15:38:1),(12:28:1),(38:28:1),(36:28:1),$$
$$(40:41:1),(10:25:1),(20:25:1),(13:25:1),(20:32:1),(42:10:1),$$
$$(35:10:1),(9:10:1),(40:30:1),(13:30:1),(33:30:1),(24:4:1),$$
$$(25:36:1),(12:36:1),(6:36:1),(12:22:1),(14:23:1),(8:23:1),$$
$$(21:23:1),(24:26:1),(37:26:1),(25:26:1),(23:35:1),(15:14:1),$$
$$(33:14:1),(38:14:1),(33:42:1),(17:40:1),(4:40:1),(22:40:1),$$
$$(5:34:1),(15:34:1),(23:34:1),(31:16:1),(40:15:1),(37:15:1),$$
$$(9:15:1),(37:2:1),(11:6:1),(39:6:1),(36:6:1),(31:18:1),(9:18:1),$$
$$(3:18:1),(10:11:1),(5:13:1),(24:13:1),(14:13:1),(5:39:1),$$
$$(41:31:1),(13:31:1),(32:31:1),(10:7:1),(14:7:1),(19:7:1),$$
$$(6:21:1),(17:17:1),(3:17:1),(23:17:1),(3:8:1),(25:24:1),$$
$$(16:24:1),(2:24:1),(20:29:1),(17:29:1),(6:29:1),(38:1:1)\}.$$

The orbit of $(1:0:0) \in X(\mathbb{F})$ under G is the following set of 24 points:

$$\{(1:0:0),(0:1:0),(0:0:1),(2:24:1),(3:8:1),(5:39:1),(8:23:1),$$
$$(9:18:1),(12:22:1),(13:30:1),(14:7:1),(15:34:1),(17:29:1),$$
$$(19:27:1),(20:32:1),(22:40:1),(25:26:1),(27:38:1),(32:31:1),$$
$$(33:42:1),(36:28:1),(37:2:1),(39:6:1),(42:10:1)\}.$$

SAGE

```
sage: x,y,z = PolynomialRing(GF(43), 3, 'xyz').gens()
sage: X = Curve(x^3*y + y^3*z + z^3*x)
sage: X.genus()
3
sage: pts = X.rational_points(algorithm="bn")
sage: len(pts)
80
sage: inds =
[0,1,2,4,5,11,15,18,23,27,29,33,37,41,44,46,54,56,60,63,67,68,74,79]
sage: ninds = [i for i in range(80) if not(i in inds)]
sage: orbit = [pts[i] for i in inds]; orbit

[(0 : 0 : 1),
 (0 : 1 : 0),
 (1 : 0 : 0),
 (2 : 24 : 1),
 (3 : 8 : 1),
 (5 : 39 : 1),
 (8 : 23 : 1),
 (9 : 18 : 1),
 (12 : 22 : 1),
```

```
 (13 : 30 : 1),
 (14 : 7 : 1),
 (15 : 34 : 1),
 (17 : 29 : 1),
 (19 : 27 : 1),
 (20 : 32 : 1),
 (22 : 40 : 1),
 (25 : 26 : 1),
 (27 : 38 : 1),
 (32 : 31 : 1),
 (33 : 42 : 1),
 (36 : 28 : 1),
 (37 : 2 : 1),
 (39 : 6 : 1),
 (42 : 10 : 1)]
sage: supp = [ (1, pts[i]) for i in inds]
sage: D = X.divisor(supp)
sage: basis = X.riemann_roch_basis(D)
sage: len(basis)
22
```

We evaluate each element of the Riemann–Roch space $L(D)$ at a point in the complement of the above-mentioned orbit in the set of rational points $X(\mathbb{F})$.

SAGE

```
sage: cfts = [[f(pts[i][0],pts[i][1],pts[i][2]) for i in ninds] for f in basis]
sage: G = matrix(cfts)
sage: C = LinearCode(G); C
Linear code of length 56, dimension 22 over Finite Field of size 43
```

With $r = 1$, we expect that $C = C(X, D, E)$ is a $[56, 22, 32]$ code.[8] With $r = 2$, we expect that $C = C(X, D, E)$ is a $[56, 46, 8]$ code. With $r = 3$, $C = C(X, D, E)$ is a $[56, 56, 1]$ code. In case $r = 1, 2$, eval_E is injective, but when $r = 3$, it is not. Indeed, $\dim L(3D_\infty) = 70$. In each case, the dimension of C can be computed using SAGE (and Singular), but the minimum distance cannot.

Remark 26

- Indeed, it is known more generally that, for an AG code constructed as above from a curve of genus g, $n \leq \dim(C) + d(C) + g - 1$, where $d(C)$ denotes the minimum distance (Theorem 3.1.1 in [TV] or Lemma 189 in Sect. 6.5 below). Therefore, as an AG code, the codes constructed above with $r = 1, 2$ are in some sense "best possible."
- In general, Sect. 4.1 of [TV] shows how to construct a family of "good" codes from the curves $X = X'_N$ for prime $N > 5$, with automorphism group $G = PSL(2, p)$.

[8] In other words, C has length 56, dimension 22 over $\mathbb{F}$, and minimum distance 32.

Table 6.1 Models of genus 1 modular curves

Level	Discriminant	Weierstrass model	Reference
11	-11	$y^2+y=x^3-x^2$	[BK], Table 1, p. 82
14	-28	$y^2+xy-y=x^3$	p. 391, Table 12.1 of [Kn]
15	15	$y^2+7xy+2y=x^3+4x^2+x$	p. 65, Table 3.2 of [Kn]
17	17	$y^2+3xy=x^3+x$	p. 65, Table 3.2 of [Kn]
19	-19	$y^2+y=x^3+x^2+x$	[BK], Table 1, p. 82
20	80	$y^2=x^3+x^2-x$	p. 391, Table 12.1 of [Kn]
21	-63	$y^2+xy=x^3+x$	p. 391, Table 12.1 of [Kn]
24	-48	$y^2=x^3-x^2+x$	p. 391, Table 12.1 of [Kn]
27	-27	$y^2+y=x^3$	p. 391, Table 12.1 of [Kn]
32	64	$y^2=x^3-x$	p. 391, Table 12.1 of [Kn]
36			Sect. 4.3 in [Ro]
49			Sect. 4.3 in [Ro]

6.4.1 The Curves $X_0(N)$ of Genus 1

It is known (see, for example, [Kn]) that a modular curve of level N, $X_0(N)$, is of genus 1 if and only if

$$N \in \{11, 14, 15, 17, 19, 20, 21, 24, 27, 32, 36, 49\}.$$

In these cases, $X_0(N)$ is birational to an elliptic curve E having Weierstrass model of the form

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

with a_1, a_2, a_3, a_4, a_6. If E is of the above form, then the *discriminant* is given by

$$\Delta = -b_2^2b_8 - 8b_4^3 - 27b_6^2 + 9b_2b_4b_6,$$

where

$$b_2 = a_1^2 + 4a_2, \qquad b_4 = 2a_4 + a_1a_3, \qquad b_6 = a_3^2 + 4a_6,$$
$$b_8 = a_1^2a_6 + 2a_2a_6 - a_1a_3a_4 + a_2a_3^2 - a_4^2.$$

The conductor[9] N of E and its discriminant Δ have the same prime factors. Furthermore, $N|\Delta$ [Kn, Gel].

Some examples, which we shall use later, are collected in Table 6.1.

When $N = 36$, Sect. 4.3 in Rovira [Ro] gives $y^2 = x^4 - 4x^3 - 6x^2 - 4x + 1$, which is a hyperelliptic equation but not in Weierstrass form. When $N = 49$,

[9] The conductor is defined in Ogg [O1], but see also [Gel], Sect. I.2, or [Kn], p. 390.

Sect. 4.3 in Rovira [Ro] gives $y^2 = x^4 - 2x^3 - 9x^2 + 10x - 3$, which is a hyperelliptic equation but not in Weierstrass form.

6.5 Some Estimates on AG Codes

This currently is an active field of research. An excellent general reference is the 2010 survey paper by Li [Li]. The survey by Li also presents recent work of Elkies, Xing, Li, Maharaj, Stichtenoth, Niederreiter, Özbudak, Yang, Qi and others, with more recent advances than described here. Below, some of the basic well-known estimates are discussed.

Let g be the genus of a curve $V = X$, and let $C = C(D, E, X)$ denote the AG code as constructed above in (6.2.2). If C has parameters $[n, k, d]$, then the following lemma is a consequence of the Riemann–Roch theorem.

Lemma 189 *Assume that C is as above and D satisfies $2g - 2 < \deg(D) < n$. Then $k = \dim(C) = \deg(D) - g + 1$ and $d \geq n - \deg(D)$.*

Consequently, $k + d \geq n - g + 1$. Because of Singleton's inequality,[10] we have:

- if $g = 0$, then C is an MDS code,
- if $g = 1$, then $n \leq k + d \leq n + 1$.

The previous lemma also implies the following lower bound.

Proposition 190 ([SS], Sect. 3.1, or [TV]) *With C as in the previous lemma, we have $\delta + R = \frac{d}{n} + \frac{k}{n} \geq 1 - \frac{g-1}{n}$.*

Theorem 186 is an explicit formula for the genus of the modular curve $X_0(N)$ in terms of arithmetic data. Equation (6.3.6) is an estimate relating the genus of the modular curve with its number of points over a finite field. It may be instructive to plug these formulas into the estimate in Proposition 190 to see what we get. The formula for the genus g_N of $X_0(N)$ is relatively complicated but simplifies greatly when N is a prime number which is congruent to 1 modulo 12, say $N = 1 + 12m$, in which case $g_N = m - 1$. For example, $g_{13} = 0$. In particular, we have the following:

Corollary 191 *Let $X = X_0(N)$, where N is a prime number which is congruent to 1 modulo 12 and has the property that X is smooth over $GF(q)$. Then the parameters $[n, k, d]$ of a Goppa code associated to X must satisfy*

$$\frac{d}{n} + \frac{k}{n} \geq 1 - \frac{\frac{N-1}{12} - 2}{n}.$$

[10] Recall Singleton's bound: $n \geq d + k - 1$.

Based on Proposition 190, if one considers a family of curves X_i with increasing genus g_i such that

$$\lim_{i\to\infty} \frac{|X_i(GF(q))|}{g_i} = \alpha, \tag{6.5.1}$$

one can construct a family of codes C_i with $\delta(C_i) + R(C_i) \geq 1 - \frac{1}{\alpha}$. It is known that $\alpha \leq \sqrt{q} - 1$ (this is the so-called *Drinfeld–Vladut bound*, [TV], Theorem 2.3.22).

The following result says that the Drinfeld–Vladut bound can be attained in the case $q = p^2$.

Theorem 192 (Tsfasman, Valdut, and Zink [TV], Theorem 4.1.52) *Let g_N denote the genus of $X_0(N)$. If N runs over a set of primes different from p, then the quotients $g_N/|X_0(N)(GF(p^2))|$ associated to the modular curves $X_0(N)$ tend to the limit $\frac{1}{p-1}$.*

More generally, if $q = p^{2k}$, then there is a family of Drinfeld curves X_i over $GF(q)$ yielding $\alpha = \sqrt{q} - 1$ ([TV], Theorem 4.2.38, discovered independently by Ihara [I] at about the same time). In other words, the Drinfeld–Vladut bound is attained in the case $q = p^{2k}$.

As a corollary to the above theorem, if $p \geq 7$, then there exists a sequence of AG codes C_N over $GF(p^2)$ associated to a sequence of modular curves $X_0(N)$ for which $(R(C_N), \delta(C_N))$ eventually (for suitable large N) lies above the Gilbert–Varshamov bound in Theorem 21. This follows from comparing the Gilbert–Varshamov curve

$$\big(\delta, f_q(\delta)\big),$$
$$f_q(\delta) = 1 - \delta \log_q\left(\frac{q-1}{q}\right) - \delta \log_q(\delta) - (1-\delta)\log_q(1-\delta),$$

with the curve $(\delta, \frac{1}{\sqrt{q}-1})$, $q = p^2$.

6.6 Examples

Let X be an elliptic curve. This is a projective curve for which $X(GF(q))$ has the structure of an algebraic group. Let $P_0 \in X(GF(q))$ denote the identity. Let $P_1, P_2, \ldots, P_n$ denote all the other elements of $X(GF(q))$, and let $A = aP_0$, where $0 < a < n$ is an integer.

Example 193 Let X denote the elliptic curve of conductor 32 (and birational to $X_0(32)$) with Weierstrass form $y^2 = x^3 - x$. Let $X(GF(p)) = \{P_0, P_1, P_2, \ldots, P_n\}$, where P_0 is the identity, and let $D = kP_0$ for some $k > 0$, $E = P_1 + \cdots + P_n$. If p is a prime satisfying $p \equiv 3 \pmod 4$, then

$$\big|X\big(GF(p)\big)\big| = p + 1$$

(Theorem 5, Sect. 18.4 in Ireland and Rosen [IR]). The parameters of the corresponding code $C = C(D, E, X)$ satisfy $n = p$ and $d + k \geq n$, since $g = 1$ by the above proposition. As we observed above, an AG code constructed from an elliptic curve satisfies either $d + k - 1 = n$ (i.e., is MDS) or $d + k = n$. The result of Shokrollahi below implies that if, in addition, $p > 3$ or $k > 2$, then C is not MDS, and

$$n = p, \qquad d + k = p.$$

The following result is an immediate corollary of the results in [Sh1], see also Sect. 5.2.2 in [TV].

Theorem 194 (Shokrollahi) *Let $X, P_0, P_1, \ldots, P_n, D, E$, be as above.*

- *If $a = 2$ and $X(GF(q)) \cong C_2 \times C_2$ (where C_n denotes the cyclic group of order n), then the code $C = C(D, E)$ is an $[n, k, d]$-code (n is the length, k is the dimension, and d is the minimum distance) with*

$$d = n - k + 1 \quad \text{and} \quad k = a.$$

- *Assume that* $\gcd(n, a!) = 1$. *If $a \neq 2$ or $X(GF(q))$ is not isomorphic to the Klein four group $C_2 \times C_2$, then $C = C(D, E)$ is an $[n, k, d]$-code (n is the length, k is the dimension, and d is the minimum distance) with*

$$k = a$$

and weight enumerator polynomial (see, for example, [MS] for the definition)

$$W_C(x) = x^n + \sum_{i=0}^{a-1} \binom{n}{i} (q^{a-i} - 1)(x - 1)^i + B_a(x - 1)^a,$$

where B_a is given in [Sh1] and Sect. 3.2.2 in [TV].

6.6.1 The Generator Matrix (According to Goppa)

This section uses the method of Goppa's book [G1] to compute the generator matrices of some AG codes.

Example 195 Consider the hyperelliptic curve[11] X defined by $y^2 = x^p - x$ over the field $GF(p)$ with p elements. It is easy to see that

$$X\big(GF(p)\big) = \big\{P_\infty, (0, 0), (1, 0), \ldots, (p - 1, 0)\big\}$$

[11] When $p = 3$ it is a model of a modular curve of level 32 (see Table 6.1). When $p = 7$ this example arises in the reduction of $X(7)$ in characteristic 7 [E2].

has exactly $p+1$ points, including the point at infinity, P_∞. The automorphism group of this curve is a twofold cover of $PSL(2,p)$ (see Göb [Go] for the algebraically closed case).

Consider, for example, the case of $p=7$. Let $D=mP_\infty$ and $E=P_1+\cdots+P_7$, and let C denote the one-point AG code associated to $X/GF(7)$ and these divisors D, E. These codes give rise to MDS codes in many cases.

When $m=2$, we obtain a $[7,2,6]$ code with weight enumerator $1+42x^6+6x^7$. This code has automorphism group of order 252 and permutation group of order 42. When $m=4$, we obtain a $[7,3,5]$ code with weight enumerator $1+126x^5+84x^6+132x^7$. This code has the same automorphism group and permutation group. It has the generator matrix in standard form

$$G=\begin{pmatrix}1&0&0&2&5&1&5\\0&1&0&1&5&5&2\\0&0&1&5&5&2&1\end{pmatrix}$$

and check matrix

$$H=\begin{pmatrix}5&6&2&1&0&0&0\\2&2&2&0&1&0&0\\6&2&5&0&0&1&0\\2&5&6&0&0&0&1\end{pmatrix}.$$

The method used in Goppa's Fermat cubic code example of [G1] (pp. 108–109) can be easily modified to yield analogous quantities for certain elliptic Goppa codes.

Example 196 Let X denote the elliptic curve (of conductor $N=19$) which we write in homogeneous coordinates as

$$y^2z+yz^2=x^3+x^2z+xz^2.$$

Let $\phi(x,y,z)=x^2+y^2+z^2$, let Y denote the projective curve defined by $\phi(x,y,z)=0$, and let D denote the divisor obtained by intersecting X and Y. By Bezout's theorem, D is of degree 6. A basis for $\mathcal{L}(D)$ is provided by the functions in the set

$$\mathcal{B}_D=\left\{1,x^2/\phi(x,y,z),y^2/\phi(x,y,z),z^2/\phi(x,y,z),xy/\phi(x,y,z),yz/\phi(x,y,z)\right\}.$$

(This is due to the fact that $\dim\mathcal{L}(D)=\deg(D)=6$ and the functions $f\in\mathcal{B}_D$ "obviously" satisfy $(f)\geq -D$.) We have

$$X\big(GF(7)\big)=\big\{[0,0,1],[0,1,0],[0,1,6],[1,0,2],[1,0,4],[1,3,4],[1,3,6],\\[1,5,2],[1,5,6]\big\},$$

which we write as $P_1, P_2, \ldots, P_9$. Consider the matrix

$$G = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 2 & 2 & 4 & 4 \\ 1 & 0 & 1 & 4 & 2 & 2 & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 3 & 3 & 5 & 5 \\ 0 & 0 & 6 & 0 & 0 & 5 & 4 & 3 & 2 \\ 0 & 0 & 0 & 2 & 4 & 4 & 6 & 2 & 6 \end{bmatrix}.$$

The first row of G gives the values of $x^2/\phi(x, y, z)$ at $\{P_i | 1 \leq i \leq 9\}$. The other rows are obtained similarly from the other functions corresponding to the basis elements of $\mathcal{L}(D)$: $y^2/\phi(x, y, z)$, $z^2/\phi(x, y, z)$, $xy/\phi(x, y, z)$, $yz/\phi(x, y, z)$. Performing Gauss reduction mod 7 puts this in canonical form:

$$G' = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 4 \\ 0 & 1 & 0 & 0 & 0 & 0 & 6 & 0 & 6 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 3 & 4 \\ 0 & 0 & 0 & 1 & 0 & 0 & 6 & 1 & 6 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 4 & 4 \end{bmatrix},$$

so this code also has minimum distance 3 and hence is only 1-error correcting. The corresponding check matrix is

$$H = \begin{bmatrix} 0 & 1 & 2 & 1 & 2 & 2 & 1 & 0 & 0 \\ 3 & 0 & 4 & 6 & 4 & 3 & 0 & 1 & 0 \\ 3 & 1 & 3 & 1 & 2 & 3 & 0 & 0 & 1 \end{bmatrix}.$$

An example of the generating matrix of a one-point elliptic code associated to $x^3 + y^3 = 1$ over $GF(4)$ has been worked out in several places (for example, see Goppa's book mentioned above or the books [SS], Sect. 3.3, [P], Sects. 5.3, 5.4, 5.7, and [Mo], Sect. 5.7.3).

6.7 Ramification Module of $X(N)$

The following result is due to Joyner and Ksir [JK1]. We use the notation of (6.1.1) and of the appendix Sect. 7.5 below.

Theorem 197 *We have the following decomposition of the ramification module*:

$$\tilde{\Gamma}_G = \sum_{\pi} m_\pi(N)\pi,$$

where $m_\pi(N)$ is an explicit multiplicity which satisfies $\frac{N}{4} \leq m_\pi(N) \leq \frac{5N}{4}$ for all N.

The formula for $m_\pi(N)$ is, though explicit, fairly complicated and will not be stated here (see [JK1] for details).

Open Problem 31 Suppose that X is a smooth projective curve with (a) genus greater than 1, (b) automorphism group G, and (c) defined over a field F with "bad" characteristic p (that is, p divides the order of G). Is there an analog of Theorem 197?

Is there an $\mathbb{F}[G]$-module decomposition of an arbitrary AG code analogous to (6.1.1)?

If $\tilde{\Gamma}_G$ has a $\mathbb{Q}[G]$-module structure, it may be computed more simply. In this case, the formula for it is

$$\tilde{\Gamma}_G = \bigoplus_{\pi \in G^*} \left[\sum_{\ell=1}^{L} \left(\dim \pi - \dim\left(\pi^{H_\ell}\right)\right) \frac{R_\ell}{2} \right] \pi, \tag{6.7.1}$$

where $\{H_1, \ldots, H_L\}$ represent the set of conjugacy classes of cyclic subgroups of G [JK2]. If $\tilde{\Gamma}_G$ does not have a natural $\mathbb{Q}[G]$-module structure, then the situation is more complicated, and we refer the reader to[JK1] for more details.

This motivates the following, as stated in the introduction.

Theorem 198 *For $N > 5$ prime, the ramification module of $X(N)$ over $X(1)$ has a natural $\mathbb{Q}[G]$-module structure if and only if $N \equiv 1 \pmod 4$.*

In this case, we can use formula (6.7.1) to compute the ramification module directly from the restricted representations (for details, see [JK1]). If $N \equiv 3 \pmod 4$, the situation is more complicated, but we refer to [JK1] for details.

6.7.1 Example: $N = 7$

The texts Fulton and Harris [FH] and Serre [Se2] are good general references for (complex) representations over finite groups.

The computer algebra system [GAP] computes information about $PSL(2, N)$; one can use it to compute character tables, induced characters, and Schur inner products (the computations can also be done in SAGE). In the examples of $X(7)$ and $X(11)$ below, we use (6.1.1) to explicitly compute the G-module structure of the ramification module and some Riemann–Roch spaces in the case $N = 7$.

The equivalence classes of irreducible representations of $PSL(2, 7)$ are $G^* = \{\pi_1, \pi_2, \ldots, \pi_6\}$, where

$$\dim(\pi_1) = 1, \qquad \dim(\pi_2) = \dim(\pi_3) = 3, \qquad \dim(\pi_4) = 6,$$
$$\dim(\pi_5) = 7, \qquad \dim(\pi_6) = 8.$$

Let $\zeta = e^{\frac{2\pi i}{7}}$, and let $\mathbb{Q}(q)$ denote the (quadratic) extension of $\mathbb{Q}$ by $q = \zeta + \zeta^2 + \zeta^4$. Let $\mathcal{G}$ denote the Galois group of $\mathbb{Q}(q)/\mathbb{Q}$. Then $\mathcal{G}$ acts on the irreducible representations G^* by swapping the two three-dimensional representations and fixing the others.

There are four conjugacy classes of nontrivial cyclic subgroups of G, whose representatives are denoted by H_1 (order 2), H_2 (order 3), H_3 (order 7), H_4 (order 4). We use GAP to compute the induced characters:

- If $\theta_1 \in H_1^*$, then $\pi_{\theta_1} = \mathrm{Ind}_{H_1}^G \theta_1$ is 84-dimensional. Moreover,

$$\pi_{\theta_1} \cong \begin{cases} 2\pi_2 \oplus 2\pi_3 \oplus 2\pi_4 \oplus 4\pi_5 \oplus 4\pi_6, & \theta_1 \neq 1, \\ \pi_1 \oplus \pi_2 \oplus \pi_3 \oplus 4\pi_4 \oplus 3\pi_5 \oplus 4\pi_6, & \theta_1 = 1. \end{cases}$$

- If $\theta_2 \in H_2^*$, then $\pi_{\theta_2} = \mathrm{Ind}_{H_2}^G \theta_2$ is 56-dimensional. Moreover,

$$\pi_{\theta_2} \cong \begin{cases} \pi_2 \oplus \pi_3 \oplus 2\pi_4 \oplus 2\pi_5 \oplus 3\pi_6, & \theta_2 \neq 1, \\ \pi_1 \oplus \pi_2 \oplus \pi_3 \oplus 2\pi_4 \oplus 3\pi_5 \oplus 2\pi_6, & \theta_2 = 1. \end{cases}$$

- If $\theta_3 \in H_3^*$ is a fixed nontrivial character, then $\pi_{\theta_3} = \mathrm{Ind}_{H_3}^G \theta_3$ is 24-dimensional. Moreover,

$$\pi_{\theta_3^k} \cong \begin{cases} \pi_3 \oplus \pi_4 \oplus \pi_5 \oplus \pi_6, & k \text{ quad. nonres. } (\mathrm{mod}\ 7), \\ \pi_2 \oplus \pi_4 \oplus \pi_5 \oplus \pi_6, & k \text{ quad. res. } (\mathrm{mod}\ 7), \\ \pi_1 \oplus \pi_5 \oplus 2\pi_6, & k \equiv 0 \ (\mathrm{mod}\ 7). \end{cases}$$

This data allows us to easily compute the ramification module using equations in [JK1]:

$$\left[\tilde{\Gamma}_G\right] = [3\pi_2 \oplus 4\pi_3 \oplus 6\pi_4 \oplus 7\pi_5 \oplus 8\pi_6]. \tag{6.7.2}$$

Note that this is not Galois-invariant, because π_2 and π_3 have different multiplicities. A naïve computation of the ramification module, using (6.7.1), yields the following. For brevity, we represent $m_1[\pi_1] + \cdots + m_6[\pi_6]$ as $(m_1, \ldots, m_6)$. We compute, using GAP, the quantities

$$\begin{aligned}
\left(\dim \pi - \dim\left(\pi^{H_1}\right)\right)_{i=1..6} &= (1,3,3,6,7,8) - (1,1,1,4,3,4) = (0,2,2,2,4,4), \\
\left(\dim \pi - \dim\left(\pi^{H_2}\right)\right)_{i=1..6} &= (1,3,3,6,7,8) - (1,1,1,2,3,2) = (0,2,2,4,4,6), \\
\left(\dim \pi - \dim\left(\pi^{H_3}\right)\right)_{i=1..6} &= (1,3,3,6,7,8) - (1,0,0,0,1,2) = (0,3,3,6,6,6), \\
\left(\dim \pi - \dim\left(\pi^{H_4}\right)\right)_{i=1..6} &= (1,3,3,6,7,8) - (1,1,1,2,1,2) = (0,2,2,4,6,6).
\end{aligned}$$

Combining this with $R_1 = R_2 = R_3 = 1$ and $R_4 = 0$ in (6.7.2) gives

$$\begin{aligned}
\left[\tilde{\Gamma}_G\right] &= \left[\bigoplus_{i=1}^{6} \left[\sum_{\ell=1}^{4} \left(\dim \pi_i - \dim\left(\pi_i^{H_\ell}\right)\right) \frac{R_\ell}{2}\right] \pi_i\right] \\
&= (0,2,2,2,4,4)\frac{1}{2} + (0,2,2,4,4,6)\frac{1}{2} + (0,3,3,6,6,6)\frac{1}{2} \\
&\quad + (0,2,2,4,6,6)\frac{0}{2}
\end{aligned}$$

$$= (0, 7/2, 7/2, 6, 7, 8)$$
$$= \frac{7}{2}[\pi_2] + \frac{7}{2}[\pi_3] + 6[\pi_4] + 7[\pi_5] + 8[\pi_6].$$

This is impossible, and therefore we see that $\tilde{\Gamma}_G$ does not have a $\mathbb{Q}[G]$-module structure in this case.

Now we will use GAP to compute the equivariant degree and Riemann–Roch module for some example divisors. Any effective G-invariant divisor on $X(7)$ will be nonspecial. Since $X(1)$ is genus zero, for $N = 7$, Borne's formula (6.1.1) becomes

$$\begin{aligned}\big[L(D)\big] &= [\pi_1 \oplus 3\pi_2 \oplus 3\pi_3 \oplus 6\pi_4 \oplus 7\pi_5 \oplus 8\pi_6] + \big[\deg_{eq}(D)\big] - \big[\tilde{\Gamma}_G\big] \\ &= [\pi_1 \oplus 3\pi_2 \oplus 3\pi_3 \oplus 6\pi_4 \oplus 7\pi_5 \oplus 8\pi_6] + \big[\deg_{eq}(D)\big] \\ &\quad - 3[\pi_2] - 4[\pi_3] - 6[\pi_4] - 7[\pi_5] - 8[\pi_6] \\ &= [\pi_1] - [\pi_3] + \big[\deg_{eq}(D)\big].\end{aligned}$$

If D_1 is the reduced orbit of a point with stabilizer H_1, then

$$\big[\deg_{eq}(D_1)\big] = [\pi_{\theta_1}] = [2\pi_2 \oplus 2\pi_3 \oplus 2\pi_4 \oplus 4\pi_5 \oplus 4\pi_6]$$

and

$$\big[L(D_1)\big] = [\pi_1 \oplus 2\pi_2 \oplus \pi_3 \oplus 2\pi_4 \oplus 4\pi_5 \oplus 4\pi_6].$$

If D_2 is the reduced orbit of a point with stabilizer H_2, then

$$\big[\deg_{eq}(D_2)\big] = [\pi_{\theta_2}] = [\pi_2 \oplus \pi_3 \oplus 2\pi_4 \oplus 2\pi_5 \oplus 3\pi_6],$$
$$\big[\deg_{eq}(2D_2)\big] = [2\pi_{\theta_2}] = [2\pi_2 \oplus 2\pi_3 \oplus 4\pi_4 \oplus 4\pi_5 \oplus 6\pi_6],$$

and

$$\big[L(D_2)\big] = [\pi_1 \oplus \pi_2 \oplus 2\pi_4 \oplus 2\pi_5 \oplus 3\pi_6],$$
$$\big[L(2D_2)\big] = [\pi_1 \oplus 2\pi_2 \oplus \pi_3 \oplus 4\pi_4 \oplus 4\pi_5 \oplus 6\pi_6].$$

If D_3 is the reduced orbit of a point with stabilizer H_3, then

$$\big[\deg_{eq}(D_3)\big] = [\pi_{\theta_3^{N-1}}] = [\pi_3 \oplus \pi_4 \oplus \pi_5 \oplus \pi_6],$$
$$\big[\deg_{eq}(2D_3)\big] = 2[\pi_3 \oplus \pi_4 \oplus \pi_5 \oplus \pi_6],$$
$$\big[\deg_{eq}(3D_3)\big] = [\pi_2 + 2\pi_3] + 3[\pi_4 \oplus \pi_5 \oplus \pi_6],$$
$$\big[\deg_{eq}(4D_3)\big] = [\pi_2 + 3\pi_3] + 4[\pi_4 \oplus \pi_5 \oplus \pi_6],$$
$$\big[\deg_{eq}(5D_3)\big] = [2\pi_2 + 3\pi_3] + 5[\pi_4 \oplus \pi_5 \oplus \pi_6],$$
$$\big[\deg_{eq}(6D_3)\big] = [3\pi_2 + 3\pi_3] + 6[\pi_4 \oplus \pi_5 \oplus \pi_6].$$

It follows that

$$\left[L(D_3)\right] = [\pi_1] - [\pi_3] + [\pi_3 \oplus \pi_4 \oplus \pi_5 \oplus \pi_6] = [\pi_1 \oplus \pi_4 \oplus \pi_5 \oplus \pi_6],$$

which is of dimension 22, and

$$\left[L(2D_3)\right] = [\pi_1] - [\pi_3] + 2[\pi_3 \oplus \pi_4 \oplus \pi_5 \oplus \pi_6] = [\pi_1 \oplus \pi_3] + 2[\pi_4 \oplus \pi_5 \oplus \pi_6],$$

which is of dimension 46.

Chapter 7
Appendices

7.1 Coding Theory Commands in SAGE

SAGE is a mathematical software package similar to the "big Ms" (Maple, Mathematica, Magma, and Matlab) but free and open source. Download information and manuals are available at http://www.sagemath.org/. In particular, there is an excellent tutorial at http://www.sagemath.org/doc/tutorial/ (available online as a downloadable pdf).

SAGE has both a command-line interface (CLI) and a graphical user interface (GUI). For the CLI, you type in each command at a SAGE prompt `sage:` and hit return. For instance, the examples in this text have used the CLI. You can try out the SAGE GUI (sometimes called the "SAGE notebook") online at the webpage http://www.sagenb.org.

Included, among other things, in SAGE is the group theory package GAP. In addition, SAGE has many optional packages which are easy to load (if you are online) using SAGE's `install_package` command. For example, GUAVA, GAP's coding theory package, can be easily installed into SAGE using the command `install_package(gap_packages)`. All of GUAVA's functions can then be accessed within SAGE.

D. Joyner, J.-L. Kim, *Selected Unsolved Problems in Coding Theory*,
Applied and Numerical Harmonic Analysis,
DOI 10.1007/978-0-8176-8256-9_7, © Springer Science+Business Media, LLC 2011

Selected SAGE commands in coding theory:

General constructions	`LinearCode`, `LinearCodeFromCheckMatrix` `LinearCodeFromVectorspace` `RandomLinearCode`
Coding theory functions (general)	`spectrum`, `minimum_distance` `characteristic_function`, `binomial_moment` `gen_mat`, `check_mat`, `decode` `binomial_moment`, `chinen_polynomial` `standard_form`, `divisor`, `genus` `random_element`, `redundancy_matrix` `support`, `weight_enumerator` `zeta_polynomial`, `zeta_function`
Code constructions	`dual_code`, `extended_code` `direct_sum`, `punctured`, `shortened` `permuted_code`, `galois_closure`
Coding theory functions (boolean)	`is_self_dual`, `==` `is_self_orthogonal`, `is_subcode` `is_permutation_automorphism` `is_galois_closed`

Most of these functions are accessed using the following type of syntax.

SAGE

```
sage: C = HammingCode(3, GF(2)); C
Linear code of length 7, dimension 4 over Finite Field of size 2
sage: C.weight_enumerator()
x^7 + 7*x^4*y^3 + 7*x^3*y^4 + y^7
sage: C.is_self_dual()
False
sage: C.dual_code()
Linear code of length 7, dimension 3 over Finite Field of size 2
```

Roughly speaking, most lower-case commands, such as `dual_code`, are "methods" which are accessed by applying them to an object, such as `C`, using the "." operator. If you have any question about the syntax, use SAGE the help system. For example, by typing `HammingCode?` or `C.dual_code?`, you will see displayed a page which briefly explains the syntax and gives an example of its use. If you are online, you can also usually type "HammingCode sagemath" (without the quotes) into `google` to get a link to the online SAGE webpage for the reference manual for the command you want information on.

Coding theory functions (group theoretical)	`module_composition_factors` `permutation_automorphism_group`
Coding theory functions (combinatorial)	`assmus_mattson_designs`

Special constructions	`BinaryGolayCode`, `ExtendedBinaryGolayCode`
	`TernaryGolayCode`, `ExtendedTernaryGolayCode`
	`CyclicCodeFromGeneratingPolynomial` (= `CyclicCode`)
	`CyclicCodeFromCheckPolynomial`, `BCHCode`
	`DuadicCodeEvenPair`, `DuadicCodeOddPair`
	`HammingCode`, `QuadraticResidueCodeEvenPair`
	`QuadraticResidueCodeOddPair`, `QuadraticResidueCode`
	`ExtendedQuadraticResidueCode`, `ReedSolomonCode`
	`self_dual_codes_binary`
	`ToricCode`, `WalshCode`

Code bounds	`best_known_linear_code_www`, `bounds_minimum_distance`
	`codesize_upper_bound(n,d,q)`, `dimension_upper_bound(n,d,q)`
	`gilbert_lower_bound(n,q,d)`, `plotkin_upper_bound(n,q,d)`
	`griesmer_upper_bound(n,q,d)`, `elias_upper_bound(n,q,d)`
	`hamming_upper_bound(n,q,d)`, `singleton_upper_bound(n,q,d)`
	`gv_info_rate(n,delta,q)`, `gv_bound_asymp(delta,q)`
	`plotkin_bound_asymp(delta,q)`, `elias_bound_asymp(delta,q)`
	`hamming_bound_asymp(delta,q)`, `singleton_bound_asymp(delta,q)`
	`mrrw1_bound_asymp(delta,q)`

- SAGE also includes a database of all self-dual binary codes of length ≤ 20. The main function is `self_dual_codes_binary`, which is a case-by-case list of entries, each represented by a Python dictionary. See also Sect. 7.3 below.
- SAGE can now compute automorphism groups of binary linear codes very efficiently, thanks to work of Robert Miller [M1]. Thomas Fuelner has worked hard on extending this programming to much more general cases.
- A lot of work on the classification of doubly even self-orthogonal codes using SAGE can be found at http://www.rlmiller.org/de_codes/.

7.2 Finite Fields

Probably this section is redundant, given that most readers can be assumed to be familiar with the basic constructions and results presented here. However, it is provided for convenience in case a standard reference such as Lidl and Niederreiter [LN] is not easily accessible.

The usual "alphabet" of a code is $GF(2) = \{0, 1\}$, which we can regard as a field with two elements. Mathematically, it is often no harder to replace this alphabet with any finite field, so this section introduces some terminology and background about constructing finite fields. For details, see, for example, [MS].

The *prime fields*: If $p \geq 2$ is a prime, then $GF(p)$ denotes the field $\mathbb{Z}/p\mathbb{Z}$ with addition and multiplication performed mod p.

The *prime power fields*: Suppose that $q = p^r$ is a prime power, $r > 1$, and put $\mathbb{F} = GF(p)$. Let $\mathbb{F}[x]$ denote the ring of all polynomials over $\mathbb{F}$, and let $f(x)$ denote a monic irreducible polynomial in $\mathbb{F}[x]$ of degree r. The quotient $\mathbb{E} = \mathbb{F}[x]/(f(x)) = \mathbb{F}[x]/f(x)\mathbb{F}[x]$ is a field with q elements.[1] If $f(x)$ and $\mathbb{E}$ are related in this way, we say that $f(x)$ is the *defining polynomial* of $\mathbb{E}$. Any defining polynomial factors completely into distinct linear factors over the field it defines.

All finite fields arise from one of the above two constructions. Up to isomorphism, for each $r \geq 1$, there is only one field of order $q = p^r$. This field will be denoted $GF(q)$.

For any finite field $\mathbb{F}$, the multiplicative group of nonzero elements $\mathbb{F}^\times$ is a cyclic group. An $\alpha \in \mathbb{F}$ is called a *primitive element* if it is a generator of $\mathbb{F}^\times$. A defining polynomial $f(x)$ of $\mathbb{F}$ is said to be *primitive* if it has a root in $\mathbb{F}$ which is a primitive element.

Matrix Representation Let $\mathbb{E}$ denote a field extension of the finite field $\mathbb{F}$. Each nonzero element of $\mathbb{E}$ may be represented as an invertible matrix with entries in $\mathbb{F}$. Here is how. Let $\alpha \in \mathbb{E}$ denote a generator of the cyclic group $\mathbb{E}^\times$. Let $f(x)$ denote the minimal polynomial of α (the lowest degree monic polynomial in $\mathbb{F}[x]$ which has α as a root). Take the matrix associated to α, denoted A, to be the companion matrix of $f(x)$ (so the characteristic polynomial of A is f). If the degree of $f(x)$ is m, then A is an $m \times m$ matrix with coefficients in $\mathbb{F}$ (and the degree of $\mathbb{E}/\mathbb{F}$ is m). If $\beta \in \mathbb{E}$ denotes any other nonzero element, then we can write $\beta = \alpha^i$ for some i (because $\mathbb{E}^\times$ is a cyclic group). Take the matrix associated to β to be $B = A^i$. The matrix associated to $0 \in \mathbb{E}$ will be the $m \times m$ zero matrix. Therefore, there is a representation

$$\rho : \mathbb{E}^\times \to \mathrm{Aut}_{\mathbb{F}}\left(\mathbb{F}^m\right)$$

induced by this action of the field $\mathbb{E}$ acting on itself, regarded as (an $\mathbb{F}$-vector space identified with) $\mathbb{F}^m$.

Example 199 Taking $\mathbb{F} = GF(2)$ and $\mathbb{E} = GF(16)$ with defining polynomial $f(x) = x^4 + x^3 + 1$, we can represent the nonzero elements of $GF(16)$ as the following 15 matrices using powers of the first matrix:

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix},$$

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix},$$

[1] Intuitively, one may think of $\mathbb{F}[x]$ as an analog of $\mathbb{Z}$, $f(x)$ as an analog of a prime p, and $\mathbb{F}[x]/f(x)\mathbb{F}[x]$ as an analog of $\mathbb{Z}/p\mathbb{Z}$.

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix},$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Of course, matrix addition and multiplication corresponds to addition and multiplication of the corresponding field elements.

Conway Polynomials There is no canonical choice of $GF(q)$, but there is a "good" choice: take $f(x)$ to be the Conway polynomial over $GF(p)$ of degree r. This is the default finite field currently constructed by SAGE, GAP, and MAGMA.

We reproduce the definition on Frank Lübeck's Conway polynomials web page [Lu], which we refer to for further details and references.

A standard notation for the elements is given via the representatives $0, \ldots, p-1$ of the cosets modulo p. We order these elements by $0 < 1 < 2 < \cdots < p-1$. We introduce an ordering of the polynomials of degree r over $GF(p)$. Let $g(x) = g_r x^r + \cdots + g_0$ and $h(x) = h_r x^r + \cdots + h_0$ (by convention, $g_i = h_i = 0$ for $i > r$). Then we define $g < h$ if and only if there is an index k with $g_i = h_i$ for $i > k$ and $(-1)^{r-k} g_k < (-1)^{r-k} h_k$.

The *Conway polynomial* $f_{p,r}(x)$ for $GF(p^r)$ is the smallest polynomial of degree r with respect to this ordering such that:

- $f_{p,r}(x)$ is monic,
- $f_{p,r}(x)$ is primitive, that is, any zero is a generator of the (cyclic) multiplicative group of $GF(p^r)$,
- for each proper divisor m of r, we have that $f_{p,m}(x^{(p^r-1)/(p^m-1)}) \equiv 0 \pmod{f_{p,r}(x)}$; that is, the $(p^r-1)/(p^m-1)$th power of a zero of $f_{p,r}(x)$ is a zero of $f_{p,m}(x)$.

Example 200 With $p = 2$ and $r = 1$, we have $f_{2,1}(x) = x - 1 = x + 1$, as there is no other choice. Also, with $p = 2$ and $r = 2$, we have $f_{2,2}(x) = x^2 + x + 1$. Again, there is no choice, but note that $x^2 + x + 1$ divides $f_{2,1}(x^{(p^r-1)/(p^m-1)}) = f_{2,1}(x^3) = x^3 - 1$, as is required in the last condition above.

These polynomials are not easy to compute, but the fields $\mathbb{F}_1 = GF(p^{r_1})$, $\mathbb{F}_2 = GF(p^{r_2}), \ldots$ constructed from a sequence

$$f_{p,r_1}, f_{p,r_2}, f_{p,r_3}, \ldots \quad \text{with } r_i | r_{i+1}$$

have "nice" embedding properties.

Sounds complicated, but actually these fields are very easy to deal with using SAGE or Guava, GAP's error-correcting codes package [Gu].

SAGE

```
sage: conway_polynomial(2,1)
x + 1
sage: conway_polynomial(2,2)
x^2 + x + 1
sage: conway_polynomial(2,3)
x^3 + x + 1
sage: conway_polynomial(2,4)
x^4 + x + 1
sage: conway_polynomial(11,4)
x^4 + 8*x^2 + 10*x + 2
```

7.3 Tables of Self-dual Codes in SAGE

Tables of self-dual codes of small characteristic and short length are included, for example, in [HP1], Chap. 9. SAGE also includes a database of all self-dual binary codes of length ≤ 20. The main function is `self_dual_codes_binary`, which is a case-by-case list of entries, each represented by a Python dictionary.

Format of each entry: a Python dictionary with keys
`order autgp`, `spectrum`, `code`, `Comment`, `Type`, where

- `code`—a self-dual code C of length n, dimension $n/2$, over $GF(2)$,
- `order autgp`—order of the permutation automorphism group of C,
- `Type`—the type of C (which can be "I" or "II", in the binary case),
- `spectrum`—the spectrum $[A_0, A_1, \ldots, A_n]$,
- `Comment`—possibly an empty string.

In fact, in Table 9.10 of [HP1], the number B_n of inequivalent self-dual binary codes of length n is given:

n	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30
B_n	1	1	1	2	2	3	4	7	9	16	25	55	103	261	731

According to an entry in Sloane's Online Encyclopedia of Integer Sequences, http://www research.att.com/~njas/sequences/A003179, the next two entries are: 3295, 24147.

SAGE

```
sage: C = self_dual_codes_binary(10)["10"]
sage: C["0"]["code"] == C["0"]["code"].dual_code()
True
sage: C["1"]["code"] == C["1"]["code"].dual_code()
True
sage: len(C.keys()) # number of inequiv sd codes of length 10
2
sage: C = self_dual_codes_binary(12)["12"]
```

```
sage: C["0"]["code"] == C["0"]["code"].dual_code()
True
sage: C["1"]["code"] == C["1"]["code"].dual_code()
True
sage: C["2"]["code"] == C["2"]["code"].dual_code()
True
```

These SAGE commands simply show that the two inequivalent self-dual binary codes of length 10 and the two inequivalent self-dual binary codes of length 12 are indeed self dual.

A lot of work on the classification of doubly even self-orthogonal codes using SAGE can be found at http://www.rlmiller.org/de_codes/.

The number of permutation equivalence classes of all doubly even $[n, k]$-codes is shown in the table at http://www.rlmiller.org/de_codes/, and the list of codes so far discovered is linked from the list entries. Each link on that webpage points to a Sage object file, which when loaded (e.g., `sage: L = load('24_12_de_codes.sobj')`) is a list of matrices in standard form. The algorithm is described in [M2].

7.4 Proofs

7.4.1 MacWilliam's Identity

Theorem 201 (MacWilliams' identity) *If C is a linear code over $GF(q)$, then*

$$A_{C^\perp}(x, y) = |C|^{-1} A_C\big(x + (q-1)y, x - y\big).$$

Before proving this, recall the complete weight enumerator,

$$W_C(z_0, \ldots, z_{q-1}) = \sum_{c \in C} z_0^{s_0(c)} \cdots z_{q-1}^{s_{q-1}(c)} = \sum_{s \in \mathbb{Z}^q} T_C(s) z_0^{s_0} \cdots z_{q-1}^{s_{q-1}},$$

defined in (2.1.1). Sometimes, when it is convenient, we identify[2] the variables z_i with the variables z_{ω_i}. This enumerator is related to the Hamming weight enumerator as follows:

$$A_C(x, y) = W_C(x, y, \ldots, y).$$

Let $\{1, \alpha, \alpha^2, \ldots, \alpha^{m-1}\}$ denote a power basis of $GF(q)/GF(p)$. Let $\zeta = \zeta_p = e^{2\pi i/p}$ denote a pth root of unity. If $\beta, \gamma \in GF(q)$ are written $\beta = \beta_0 + \beta_1\alpha + \cdots +$

[2]Recall that we have indexed the finite field $GF(q)$ in some fixed way: $GF(q) = \{\omega_0, \omega_1, \ldots, \omega_{q-1}\}$ with $\omega_0 = 0$.

$\beta_{m-1}\alpha^{m-1}$ and $\gamma = \gamma_0 + \gamma_1\alpha + \cdots + \gamma_{m-1}\alpha^{m-1}$ ($\beta_i \in GF(p)$, $\gamma_j \in GF(p)$), then we define the character $\chi_\beta : GF(q) \to \mathbb{C}^\times$ by

$$\chi_\beta(\gamma) = \zeta^{\beta_0\gamma_0+\cdots+\beta_{q-1}\gamma_{q-1}}.$$

Each character χ of $GF(q)$ is of this form, $\chi = \chi_\beta$, for some unique $\beta \in GF(q)$. In particular,

$$\chi_1(\gamma) = \zeta^{\gamma_0}$$

for all $\gamma \in GF(q)$ (hence also for $\gamma \in GF(p)$). For $u, v \in GF(q)^n$, define

$$\chi_u(v) = \chi_1(u \cdot v),$$

and define the *Fourier transform* by

$$\hat{f}(u) = \sum_{v \in GF(q)^n} \chi_u(v) f(v)$$

for any function f on $GF(q)^n$. If $u = (u_1, \ldots, u_n) \in GF(q)^n$ and $v = (v_1, \ldots, v_n) \in GF(q)^n$, then $\chi_u(v) = \prod_{i=1}^n \chi_{u_i}(v_i)$ This means that if $f(u) = \prod_{I=1}^n f_i(u_i)$ is a "factorizable" function, then

$$\hat{f}(u) = \prod_{i=1}^n \hat{f}_i(u_i).$$

Lemma 202 (Poisson's summation formula) *If C is an $[n, k]$ code over $GF(q)$, then*

$$\sum_{c \in C^\perp} f(c) = \frac{1}{|C|} \sum_{c \in C} f(c).$$

Now we can start with proof of the MacWilliams identity. Define

$$f(u) = z_0^{s_0(u)} \cdots z_{q-1}^{s_{q-1}(u)},$$

so

$$f(u) = z_0^{s_{0,i}} \cdots z_{q-1}^{s_{q-1,i}} = \prod_{i=1}^n f_i(u_i),$$

where $s_{k,i} = 1$ if $u_i = \omega_k$ and $= 0$ otherwise. Another way to define $f(u)$ is as follows:

$$f(u) = \prod_{i=1}^n z_{u_i} = \prod_{i=1}^n f_i(u_i).$$

We then have

$$\begin{aligned}\hat{f}(u) &= \sum_{v \in GF(q)^n} \chi_u(v) f(v) \\ &= \prod_{i-1}^{n} \hat{f}_i(u_i) \\ &= \prod_{i-1}^{n} \big(F_q \cdot (z_0, \ldots, z_{q-1})\big)_i,\end{aligned}$$

where F_q is a $q \times q$ circulant matrix of elements of $GF(q)$, and $(F_q \cdot z)_i$ denotes its ith component:

$$\big(F_q \cdot (z_0, \ldots, z_{q-1})\big)_i = \sum_{\omega \in GF(q)} \chi_{\omega_i}(\omega) z_\omega = \sum_{\omega \in GF(q)} \chi_1(\omega_i \omega) z_\omega.$$

Poisson's summation formula implies

$$W_{C^\perp}(z_0, \ldots, z_{q-1}) = \frac{1}{|C|} W_C\big(F_q \cdot (z_0, \ldots, z_{q-1})\big).$$

Let $x = z_0$ and $y = z_1 = \cdots = z_{q-1}$. It remains to note that

$$\sum_{b=1}^{q-1} \chi_a(b) = \begin{cases} q-1 & \text{if } a = 0, \\ -1 & \text{if } a \neq 0. \end{cases}$$

□

7.4.2 Mallows–Sloane–Duursma Bounds

We sketch a proof of Theorem 91 following Duursma [D3]. We shall restrict to the Type 1 case for simplicity. (The Type 2 case is similar, but follow modifications as indicated in [D3], Sect. 2.) We shall also assume that F contains the term y^n (the analog of the assumption that C contains the all-ones codeword).

Some notation. If $p(x, y) \in \mathbb{C}[x, y]$, then we define

$$p(x, y)(D) = p\left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right).$$

If σ is any invertible 2×2 matrix and $(u, v) = (x, y)\sigma$, then

$$p\big((u, v)\sigma^t\big)(D) F(u, v) = p(x, y)(D) F\big((x, y)\sigma\big). \tag{7.4.1}$$

Lemma 203 *Fix a homogeneous function $f(x, y)$. For all i with $1 \leq i \leq e$, let $a_i \neq 0$, b_i, $c_i \neq 0$, and d_i be complex numbers satisfying $(a_i x + b_i y)^m \mid$*

$(c_i x + d_i y)(D) f(x, y)$ *for some integer* $m > 1$. *Then*

$$\prod_{i=1}^{e} (a_i x + b_i y)^{m-e+1} \Bigg| \left(\prod_{i=1}^{e} c_i x + d_i y \right)(D) f(x, y).$$

Note: This implies the result that Duursma claims in [D3], (5) (where his equation, in his special case, has $m = d^{\perp} - 1$ and $e = c$).

Proof Using (7.4.1), we may assume without loss of generality that $d_i = 0$ for all i after making a suitable linear change of coordinates. Under the coordinate transformation $z = x/y$, $f(x, y)$ may be regarded as a polynomial $F(z)$ on $\mathbb{P}^1$. If $f(x, y) = x^k y^{n-k}$, then $F(x) = z^k$, and an explicit calculation allows one to check that $D_x f(x, y)$ corresponds to $D_z F(z)$.

The hypothesis $(a_i x + b_i y)^m \mid (c_i x)(D) f(x, y)$ can be rephrased as saying that the derivative of F has roots of multiplicity m at certain points. Therefore the eth-order derivative of F gives a function which has zeros at these points of order at least $m - e + 1$. □

Let F be any virtually self-dual weight enumerator of length n and minimum distance d.

Note that if F is as above, then

$$y^{d-1} \mid y(D) F(x, y). \tag{7.4.2}$$

Equations (7.4.1) and (7.4.2) imply

$$(u - v)^{d-1} \mid \big((q-1)u - v\big)(D) F(u, v).$$

Taking $u = x$ and $v = \zeta y$, we have

$$(x - \zeta y)^{d-1} \mid \big((q-1)x - \zeta y\big)(D) F(x, y).$$

By Lemma 203, this implies

$$\big(x^b - y^b\big)^{d-b} \mid \big((q-1)^b x^b - y^b\big)(D) F(x, y). \tag{7.4.3}$$

Using (7.4.2) and (7.4.3) and reasoning similar to that used in the proof of Lemma 203, we obtain

$$a(x, y) \mid p(x, y)(D) F(x, y),$$

where $a(x, y) = y^{d-b-1}(x^b - y^b)^{d-b-1}$ and $p(x, y) = y((q-1)^b x^b - y^b)$. Comparing highest-order terms in y in $p(x, y)(D) F(x, y)$ (recall that we assumed that F contains y^n), we obtain $d - b - 1 + b(d - b - 1) \leq n - b - 1$. From this (4.3.1) follows in the Type 1 case. This is the first part of Theorem 91. The second part follows from the first by using properties of the greatest integer function in a straightforward way. □

7.5 Ramification Module and Equivariant Degree

This appendix recalls some terms for completeness. See [JK2] or [Bo] for further details.

Let X be a smooth projective curve of genus > 1 defined over an algebraically closed field k with automorphism group G.

For any point $P \in X(k)$, let G_P be the decomposition group at P (i.e., the subgroup of G fixing P). If $\mathrm{char}(k)$ does not divide $|G|$, the quotient $\pi : X \to Y = X/G$ is tamely ramified, and this group G_P is cyclic. G_P acts on the cotangent space of $X(k)$ at P by a k-character. This character is the *ramification character* of X at P.

The *ramification module* is defined by

$$\Gamma_G = \sum_{P \in X(k)_{ram}} \mathrm{Ind}_{G_P}^G \left(\sum_{\ell=1}^{e_P - 1} \ell \psi_P^\ell \right),$$

where e_P is the size of the stabilizer group, and ψ_P is the ramification character at P. By a result of Nakajima [N], there is a unique G-module $\tilde{\Gamma}_G$ such that

$$\Gamma_G = |G| \tilde{\Gamma}_G.$$

We abuse terminology and also call $\tilde{\Gamma}_G$ the *ramification module*.

Now consider a G-equivariant divisor D on $X(k)$. If $D = \frac{1}{e_P} \sum_{g \in G} g(P)$, then we call D a *reduced orbit*. The reduced orbits generate the group of G-equivariant divisors $\mathrm{Div}(X)^G$.

Definition 204 The *equivariant degree* is a map from $\mathrm{Div}(X)^G$ to the Grothendieck group $R_k(G) = \mathbb{Z}[G_k^*]$ of virtual k-characters of G,

$$\deg_{eq} : \mathrm{Div}(X)^G \to R(G),$$

defined by the following conditions:

1. $\deg_{eq}$ is additive on G-equivariant divisors of disjoint support.
2. If $D = r \frac{1}{e_P} \sum_{g \in G} g(P)$ is an orbit, then

$$\deg_{eq}(D) = \begin{cases} \mathrm{Ind}_{G_P}^G (\sum_{\ell=1}^{r} \psi^{-\ell}) & \text{if } r > 0, \\ -\mathrm{Ind}_{G_P}^G (\sum_{\ell=0}^{-(r+1)} \psi^{\ell}) & \text{if } r < 0, \\ 0 & \text{if } r = 0, \end{cases}$$

where $\psi = \psi_P$ is the ramification character of X at P.

Note: In general, this is not additive (except on those divisors which are pullbacks via π).

If $D = \pi^*(D_0)$ is the pull-back of a divisor $D_0 \in \mathrm{Div}(Y)$, then $\deg_{eq}(D)$ has a very simple form. In this case, r is a multiple of e_P, so the equivariant degree on each orbit is r/e_P times $\mathrm{Ind}_{G_P}^{G}$ of the regular representation of G_P. In this case, we have

$$\deg_{eq}(D) = \deg(D_0) \cdot \big[k[G]\big]. \tag{7.5.1}$$

References

[A1] Adler, A.: The Mathieu group M_{11} and the modular curve X_{11}. Proc. Lond. Math. Soc. **74**, 1–28 (1997)

[A2] Adler, A.: Some integral representations of $PSL_2(\mathbb{F}_p)$ and their applications. J. Algebra **72**, 115–145 (1981)

[Af] Aftab, Cheung, Kim, Thakkar, Yeddanapudi: Information theory. Student term project in a course at MIT http://web.mit.edu/6.933/www/. Preprint (2001). Available: http://web.mit.edu/6.933/www/Fall2001/Shannon2.pdf

[An] Ancochea, G.: Zeros of self-inversive polynomials. Proc. Am. Math. Soc. **4**, 900–902 (1953)

[ASV] Anderson, N., Saff, E.B., Varga, R.S.: On the Eneström-Kakeya theorem and its sharpness. Linear Algebra Appl. **28**, 5–16 (1979)

[AK] Assmus, E. Jr., Key, J.: Designs and codes. Cambridge Univ. Press, Cambridge (1992)

[AM1] Assmus, E. Jr., Mattson, H.: On the automorphism groups of Paley-Hadamard matrices. In: Bose, R., Dowling, T. (eds.) Combinatorial Mathematics and Its Applications. Univ. of North Carolina Press, Chapel Hill (1969)

[AM2] Assmus, E.: Algebraic theory of codes, II. Report AFCRL-71-0013, Air Force Cambridge Research Labs, Bedford, MA. Preprint (1971). Available: http://handle.dtic.mil/100.2/AD718114

[Ash] Ash, R.: Information Theory. Dover, New York (1965)

[Bal] Ball, S.: On large subsets of a nite vector space in which every subset of basis size is a basis. Preprint, Dec. (2010). http://www-ma4.upc.es/~simeon/jems-mds-conj-revised.pdf

[Ba] A. Barg's Coding Theory webpage http://www.ece.umd.edu/~abarg/

[BM] Bazzi, L., Mitter, S.: Some constructions of codes from group actions. Preprint (2001). Appeared as Some randomized code constructions from group actions. IEEE Trans. Inf. Theory **52**, 3210–3219 (2006). www.mit.edu/~louay/recent/rgrpactcodes.pdf

[BDHO] Bannai, E., Dougherty, S.T., Harada, M., Oura, M.: Type II codes, even unimodular lattices, and invariant rings. IEEE Trans. Inf. Theory **45**, 1194–1205 (1999)

[BCG] Bending, P., Camina, A., Guralnick, R.: Automorphisms of the modular curve $X(p)$ in positive characteristic. Preprint (2003)

[BGT] Berrou, C., Glavieux, A., Thitimajshima, P.: Near Shannon limit error-correcting coding and decoding: Turbo-codes. Communications (1993). ICC 93. Geneva. Technical Program, Conference Record, IEEE International Conference on, vol. 2, pp. 1064–1070 (1993). Available: http://www-classes.usc.edu/engr/ee-s/568/org/Original.pdf

[Bi] Bierbrauer, J.: Introduction to Coding Theory. Chapman & Hall/CRC, New York (2005)

[B] Birch, B.J.: Some calculations of modular relations. In: Kuyk, W. (ed.) Modular Forms of One Variable, I, Proc. Antwerp Conf., 1972. Lecture Notes in Math., vol. 320. Springer, New York (1973)

[BK] Birch, B.J., Kuyk, W. (eds.): Modular forms of one variable, IV, Proc. Antwerp Conf., 1972. Lecture Notes in Math., vol. 476. Springer, New York (1975)

D. Joyner, J.-L. Kim, *Selected Unsolved Problems in Coding Theory*,
Applied and Numerical Harmonic Analysis,
DOI 10.1007/978-0-8176-8256-9, © Springer Science+Business Media, LLC 2011

[BSC] Bonnecaze, A., Calderbank, A.R., Solé, P.: Quaternary quadratic residue codes and unimodular lattices. IEEE Trans. Inf. Theory **41**, 366–377 (1995)

[BoM] Bonsall, F., Marden, M.: Zeros of self-inversive polynomials. Proc. Am. Math. Soc. **3**, 471–475 (1952). Available: http://www.ams.org/journals/proc/1952-003-03/S0002-9939-1952-0047828-8/home.html

[Bo] Borne, N.: Une formule de Riemann-Roch equivariante pour des courbes. Thesis, Univ. Bordeaux (1999). Available from: http://www.dm.unibo.it/~borne/

[BHP] Brualdi, R., Huffman, W.C., Pless, V.: Handbook of Coding Theory. Elsevier, New York (1998)

[CDS] Calderbank, A., Delsarte, P., Sloane, N.: A strengthening of the Assmus–Mattson theorem. IEEE Trans. Inf. Theory **37**, 1261–1268 (1991)

[Cas1] Casselman, W.: On representations of GL_2 and the arithmetic of modular curves. In: Modular Functions of One Variable, II, Proc. Internat. Summer School, Univ. Antwerp, Antwerp, 1972. Lecture Notes in Math., vol. 349, pp. 107–141. Springer, Berlin (1973). Errata to On representations of GL_2 and the arithmetic of modular curves. In: Modular Functions of One Variable, IV, Proc. Internat. Summer School, Univ. Antwerp, Antwerp, 1972. Lecture Notes in Math., vol. 476, pp. 148–149. Springer, Berlin (1975)

[Cas2] Casselman, W.: The Hasse–Weil ζ-function of some moduli varieties of dimension greater than one. In: Automorphic Forms, Representations and L-functions, Proc. Sympos. Pure Math., Oregon State Univ., Corvallis, Ore, 1977. Proc. Sympos. Pure Math. Part 2, vol. XXXIII, pp. 141–163. Am. Math. Soc., Providence (1979)

[Cha] Chapman, R.: Preprint sent to J. Joyner (2008)

[Char] Charters, P.: Generalizing binary quadratic residue codes to higher power residues over larger fields. Finite Fields Appl. **15**, 404–413 (2009)

[Chen] Chen, W.: On the polynomials with all there zeros on the unit circle. J. Math. Anal. Appl. **190**, 714–724 (1995)

[Ch1] Chinen, K.: Zeta functions for formal weight enumerators and the extremal property. Proc. Jpn. Acad. Ser. A Math. Sci. **81**(10), 168–173 (2005)

[Ch2] Chinen, K.: Zeta functions for formal weight enumerators and an analogue of the Mallows-Sloane bound. Preprint. Available: http://arxiv.org/pdf/math/0510182

[Ch3] Chinen, K.: An abundance of invariant polynomials satisfying the Riemann hypothesis. Preprint. Available: http://arxiv.org/abs/0704.3903

[Cl] Clozel, L.: Nombre de points des variétés de Shimura sur un corps fini (d'aprés R. Kottwitz). Seminaire Bourbaki, vol. 1992/93. Asterisque No. 216 (1993), Exp. No. 766, 4, 121–149

[Co] Cohen, P.: On the coefficients of the transformation polynomials for the elliptic modular function. Math. Proc. Camb. Philos. Soc. **95**, 389–402 (1984)

[CSi] Conway, F., Siegelman, J.: Dark Hero of the Information Age. MIT Press, New York (2005)

[Co1] Conway, J.: Hexacode and tetracode—MINIMOG and MOG. In: Atkinson, M. (ed.) Computational Group Theory. Academic Press, San Diego (1984)

[Co2] Conway, J.: On Numbers and Games (ONAG). Academic Press, San Diego (1976)

[CS1] Conway, J., Sloane, N.: Sphere Packings, Lattices and Groups, 3rd edn. Springer, Berlin (1999)

[CS2] Conway, J., Sloane, N.: Lexicographic codes: error-correcting codes from game theory. IEEE Trans. Inf. Theory **32**, 337–348 (1986)

[CS3] Conway, J.H., Sloane, N.J.A.: A new upper bound on the minimal distance of self-dual codes. IEEE Trans. Inf. Theory **36**, 1319–1333 (1990)

[C] Coy, G.: Long quadratic residue codes. USNA Mathematics Dept. Honors Project (2005–2006) (advisor Prof. Joyner)

[Cu1] Curtis, R.: The Steiner system $S(5, 6, 12)$, the Mathieu group M_{12}, and the kitten. In: Atkinson, M. (ed.) Computational Group Theory. Academic Press, San Diego (1984)

[Cu2] Curtis, R.: A new combinatorial approach to M_{24}. Math. Proc. Camb. Philos. Soc. **79**, 25–42 (1976)

[Del] Deligne, P.: Variétés de Shimura. In: Automorphic Forms, Representations and L-Functions. Proc. Sympos. Pure Math. Part 2, vol. 33, pp. 247–290 (1979)

[dLG] de Launey, W., Gordon, D.: A remark on Plotkin's bound. IEEE Trans. Inf. Theory **47**, 352–355 (2001). Available: http://www.ccrwest.org/gordon/plotkin.pdf

[DH] DiPippo, S., Howe, E.: Real polynomials with all roots on the unit circle and Abelian varieties over finite fields. J. Number Theory **78**, 426–450 (1998). Available: http://arxiv.org/abs/math/9803097

[Do] S. Dougherty webpage: Does there exist a [72, 36, 16] Type II code? (Last accessed 2009-5-7 at http://academic.scranton.edu/faculty/doughertys1/72.htm)

[Dr] Drungilas, P.: Unimodular roots of reciprocal Littlewood polynomials. J. Korean Math. Soc. **45**(3), 835–840 (2008). Available: http://www.mathnet.or.kr/mathnet/thesis_file/18_J06-382.pdf

[DL] Duflo, M., Labesse, J.-P.: Sur la formule des traces de Selberg. Ann. Sci. Ecole Norm. Super. (4) **4**, 193–284 (1971)

[D1] Duursma, I.: Combinatorics of the two-variable zeta function. In: Finite Fields and Applications. Lecture Notes in Comput. Sci., vol. 2948, pp. 109–136. Springer, Berlin (2004)

[D2] Duursma, I.: Results on zeta functions for codes. In: Fifth Conference on Algebraic Geometry, Number Theory, Coding Theory and Cryptography, University of Tokyo, January 17–19 (2003)

[D3] Duursma, I.: Extremal weight enumerators and ultraspherical polynomials. Discrete Math. **268**(1–3), 103–127 (2003)

[D4] Duursma, I.: A Riemann hypothesis analogue for self-dual codes. In: Barg, Litsyn (eds.) Codes and Association Schemes. AMS Dimacs Series, vol. 56, pp. 115–124 (2001)

[D5] Duursma, I.: From weight enumerators to zeta functions. Discrete Appl. Math. **111**(1–2), 55–73 (2001)

[D6] Duursma, I.: Weight distributions of geometric Goppa codes. Trans. Am. Math. Soc. **351**, 3609–3639 (1999)

[Dw] Dwork, B.: On the rationality of the zeta function of an algebraic variety. Am. J. Math. **82**, 631–648 (1960)

[Eb] Ebeling, W.: Lattices and Codes, 2nd edn. Vieweg, Wiesbaden (2002)

[Ei] Eichler, M.: The basis problem for modular forms and the traces of Hecke operators. In: Modular Functions of One Variable, I, Proc. Internat. Summer School, Univ. Antwerp, Antwerp, 1972. Lecture Notes in Math., vol. 320, pp. 1–36. Springer, Berlin (1973)

[E1] Elkies, N.: Elliptic and modular curves over finite fields and related computational issues. In: Buell, D., Teitelbaum, J. (eds.) Computational Perspectives on Number Theory. AMS/IP Studies in Adv. Math., vol. 7, pp. 21–76 (1998)

[E2] Elkies, N.: The Klein quartic in number theory, In: Levy, S. (ed.) The Eightfold Way: The Beauty of Klein's Quartic Curve, pp. 51–102. Cambridge Univ. Press, Cambridge (1999)

[E3] Elkies, N.: Lattices, linear codes and invariants. Not. Am. Math. Soc. **47**, 1238–1245 (2000), 1382–1391

[FR] Faifman, D., Rudnick, Z.: Statistics of the zeros of zeta functions in families of hyperelliptic curves over a finite field. Preprint. Available: http://front.math.ucdavis.edu/0803.3534

[Fe] Fell, H.J.: On the zeros of convex combinations of polynomials. Pac. J. Math. **89**, 43–50 (1980)

[FM] Frey, G., Müller, M.: Arithmetic of modular curves and applications. Preprint (1998). Available: http://www.exp-math.uni-essen.de/zahlentheorie/preprints/Index.html

[FH] Fulton, W., Harris, J.: Representation Theory: A First Course. Springer, Berlin (1991)

[Ga] Gaborit, P.: Quadratic double circulant codes over fields. J. Comb. Theory, Ser. A **97**, 85–107 (2002)

[GHKP] Gaborit, P., Cary Huffman, W., Kim, J.-L., Pless, V.: On additive codes over $GF(4)$. In: DIMACS Workshop on Codes and Association Schemes. DIMACS Series in Discrete Math and Theoretical Computer Sciences, vol. 56, pp. 135–149. AMS, Providence (2001). Available: http://www.math.louisville.edu/~jlkim/dimacs4.ps

[GZ] Gaborit, P., Zemor, G.: Asymptotic improvement of the Gilbert–Varshamov bound for linear codes. IEEE Trans. Inf. Theory **IT-54**(9), 3865–3872 (2008). Available: http://arxiv.org/abs/0708.4164

[GAP] The GAP Group: GAP—Groups, algorithms, and programming. Version 4.4.10 (2007). http://www.gap-system.org
[Gel] Gelbart, S.: Elliptic curves and automorphic representations. Adv. Math. **21**(3), 235–292 (1976)
[Go] Göb, N.: Computing the automorphism groups of hyperelliptic function fields. Preprint. Available: http://front.math.ucdavis.edu/math.NT/0305284
[G1] Goppa, V.D.: Geometry and Codes. Kluwer, Amsterdam (1988)
[G2] Goppa, V.D.: Bounds for codes. Dokl. Akad. Nauk SSSR **333**, 423 (1993)
[GPS] Greuel, G.-M., Pfister, G., Schönemann, H.: SINGULAR 3.0. A Computer Algebra System for Polynomial Computations. Centre for Computer Algebra, University of Kaiserslautern (2005). http://www.singular.uni-kl.de
[Gu] GUAVA: A coding theory package for GAP. http://www.gap-system.org/Packages/guava.html

[GHK] Gulliver, T.A., Harada, M., Kim, J.-L.: Construction of some extremal self-dual codes. Discrete Math. **263**, 81–91 (2003)
[HH] Hämäläinen, H., Honkala, I., Litsyn, S., Östergård, P.: Football pools—a game for mathematicians. Am. Math. Mon. **102**, 579–588 (1995)
[HT] Harada, T., Tagami, M.: A Riemann hypothesis analogue for invariant rings. Discrete Math. **307**, 2552–2568 (2007)
[Ha] Hartshorne, R.: Algebraic Geometry. Springer, Berlin (1977)
[He] Helleseth, T.: Legendre sums and codes related to QR codes. Discrete Appl. Math. **35**, 107–113 (1992)
[HV] Helleseth, T., Voloch, J.F.: Double circulant quadratic residue codes. IEEE Trans. Inf. Theory **50**(9), 2154–2155 (2004)
[HSS] Hedayat, A.S., Sloane, J.J.A., Stufken, J.: Orthogonal Arrays: Theory and Applications. Springer, New York (1999)
[HM] Hibino, T., Murabayashi, N.: Modular equations of hyperelliptic $X_0(N)$ and an application. Acta Arith. **82**, 279–291 (1997)
[Hil] Hill, R.: A First Course in Coding Theory. Oxford Univ Press, Oxford (1986)
[Hi] Hirschfeld, J.W.P.: The main conjecture for MDS codes. In: Cryptography and Coding. Lecture Notes in Computer Science, vol. 1025. Springer, Berlin (1995). Avaialble from: http://www.maths.sussex.ac.uk/Staff/JWPH/RESEARCH/research.html
[HK] Han, S., Kim, J.-L.: The nonexistence of near-extremal formally self-dual codes. Des. Codes Cryptogr. **51**, 69–77 (2009)
[Ho] Horadam, K.: Hadamard Matrices and Their Applications. Princeton Univ. Press, Princeton (2007)
[HLTP] Houghten, S.K., Lam, C.W.H., Thiel, L.H., Parker, J.A.: The extended quadratic residue code is the only $(48, 24, 12)$ self-dual doubly-even code. IEEE Trans. Inf. Theory **49**, 53–59 (2003)
[Hu] Huffman, W.C.: On the classification and enumeration of self-dual codes. Finite Fields Appl. **11**, 451–490 (2005)
[HP1] Huffman, W.C., Pless, V.: Fundamentals of Error-Correcting Codes. Cambridge Univ. Press, Cambridge (2003)
[Ig] Igusa, J.: On the transformation theory of elliptic functions. Am. J. Math. **81**, 436–452 (1959)
[I] Ihara, Y.: Some remarks on the number of rational points of algebraic curves over finite fields. J. Fac. Sci. Univ. Tokyo **28**, 721–724 (1981)
[IR] Ireland, K., Rosen, M.: A Classical Introduction to Modern Number Theory. Grad Texts, vol. 84. Springer, Berlin (1982)
[JV] Jiang, T., Vardy, A.: Asymptotic improvement of the Gilbert–Varshamov bound on the size of binary codes. IEEE Trans. Inf. Theory **50**, 1655–1664 (2004)
[Ja1] Janusz, G.: Simple components of $\mathbb{Q}[SL(2, q)]$. Commun. Algebra **1**, 1–22 (1974)
[Ja2] Janusz, G.J.: Overlap and covering polynomials with applications to designs and self-dual codes. SIAM J. Discrete Math. **13**, 154–178 (2000)
[Jo1] Joyner, D.: On quadratic residue codes and hyperelliptic curves. Discrete Math. Theor. Comput. Sci. **10**(1), 129–126 (2008)

[Jo2] Joyner, D.: Zeros of some self-reciprocal polynomials. Submitted to proceedings of FFT 2011 (editors: Travis Andrews, Radu Balan, John J. Benedetto, Wojciech Czaja, and Kasso A. Okoudjou), Springer–Birkhauser Applied and Numerical Harmonic Analysis (ANHA) Book Series

[JK1] Joyner, D., Ksir, A.: Modular representations on some Riemann-Roch spaces of modular curves $X(N)$. In: Shaska, T. (ed.) Computational Aspects of Algebraic Curves. Lecture Notes in Computing. World Scientific, Singapore (2005)

[JK2] Joyner, D., Ksir, A.: Decomposing representations of finite groups on Riemann-Roch spaces. Proc. Am. Math. Soc. **135**, 3465–3476 (2007)

[JKT] Joyner, D., Ksir, A., Traves, W.: Automorphism groups of generalized Reed-Solomon codes. In: Shaska, T., Huffman, W.C., Joyner, D., Ustimenko, V. (eds.) Advances in Coding Theory and Cryptology. Series on Coding Theory and Cryptology, vol. 3. World Scientific, Singapore (2007)

[JKTu] Joyner, D., Kreminski, R., Turisco, J.: Applied Abstract Algebra. Johns Hopkins Univ. Press, Baltimore (2004)

[JS] Joyner, D., Shokranian, S.: Remarks on codes from modular curves: MAPLE applications. In: Joyner, D. (ed.) Coding Theory and Cryptography: From the Geheimschreiber and Enigma to Quantum Theory. Springer, Berlin (2000). Available at http://www.opensourcemath.org/books/cryptoday/cryptoday.html

[KR] Kahane, J., Ryba, A.: The hexad game. Electron. J. Comb. **8** (2001). Available at http://www.combinatorics.org/Volume_8/Abstracts/v8i2r11.html

[Kan] Kantor, W.: Automorphism groups of Hadamard matrices. J Comb Theory **6**, 279–281 (1969). Available at http://darkwing.uoregon.edu/~kantor/PAPERS/AutHadamard.pdf

[Ked] Kedlaya, K.: Search techniques for root-unitray polynomials. Preprint (2006). Available at http://arxiv.org/abs/math.NT/0608104

[KP] Khare, C., Prasad, D.: Extending local representations to global representations. Kyoto J. Math. **36**, 471–480 (1996)

[Ki1] Kim, J.-L.: A prize problem in coding theory. In: Sala, M., Mora, T., Perret, L., Sataka, S., Traverso, T. (eds.) Gröbner Basis, Coding, and Cryptography, pp. 373–376. Springer, Berlin (2009). Available: http://www.math.louisville.edu/~jlkim/jlkim_07.pdf

[KL] Kim, J.-L., Lee, Y.: Euclidean and Hermitian self-dual MDS codes over large finite fields. J. Comb. Theory, Ser. A **105**, 79–95 (2004)

[K] Kim, S.-H.: On zeros of certain sums of polynomials. Bull. Korean Math. Soc. **41**(4), 641–646 (2004). http://www.mathnet.or.kr/mathnet/kms_tex/983297.pdf

[KiP] Kim, S.-H., Park, C.W.: On the zeros of certain self-reciprocal polynomials. J. Math. Anal. Appl. **339**, 240–247 (2008)

[Kn] Knapp, A.: Elliptic Curves, Mathematical Notes. Princeton Univ. Press, Princeton (1992)

[Kob] Koblitz, N.: Introduction to Elliptic Curves and Modular Forms. Grad. Texts, vol. 97. Springer, Berlin (1984)

[Koch] Koch, H.: On self-dual doubly even extremal codes. Discrete Math. **83**, 291–300 (1990)

[KM] Konvalina, J., Matache, V.: Palindrome-polynomials with roots on the unit circle. http://myweb.unomaha.edu/~vmatache/pdffiles/short_note_cr.pdf

[K1] Kottwitz, R.: Shimura varieties and λ-adic representations. In: Automorphic Forms, Shimura Varieties, and L-functions, vol. 1, pp. 161–209. Academic Press, San Diego (1990)

[K2] Kottwitz, R.: Points on Shimura varieties over finite fields. J. Am. Math. Soc. **5**, 373–444 (1992)

[Lab] Labesse, J.P.: Exposé VI. In: Boutot, J.-F., Breen, L., Gŕardin, P., Giraud, J., Labesse, J.-P., Milne, J.S., Soulé, C. (eds.) Variétés de Shimura et fonctions L. Publications Mathématiques de l'Université Paris VII [Mathematical Publications of the University of Paris VII], 6. Universite de Paris VII, U.E.R. de Mathematiques, Paris (1979)

[LO] Lagarias, J.C., Odlyzko, A.M.: Effective versions of the Chebotarev density theorem. In: Fröhlich, A. (ed.) Algebraic Number Fields (L-functions and Galois theory), pp. 409–464. Academic Press, San Diego (1977)

[L1] Lakatos, P.: On polynomials having zeros on the unit circle. C. R. Math. Acad. Sci. Soc. R. Can. **24**(2), 91–96 (2002)
[L2] Lakatos, P.: On zeros of reciprocal polynomials. Publ. Math. (Debr.) **61**, 645–661 (2002)
[LL1] Lakatos, P., Losonczi, L.: Self-inversive polynomials whose zeros are on the unit circle. Publ. Math. (Debr.) **65**, 409–420 (2004)
[LL2] Lakatos, P.: On zeros of reciprocal polynomials of odd degree. J. Inequal. Pure Appl. Math. **4**(3) (2003). http://jipam.vu.edu.au
[Lan1] Langlands, R.P.: Shimura varieties and the Selberg trace formula. Can. J. Math. **XXIX**(5), 1292–1299 (1977)
[Lan2] Langlands, R.P.: On the zeta function of some simple Shimura varieties. Can. J. Math. **XXXI**(6), 1121–1216 (1979)
[LM] Laywine, C.F., Mullen, G.L.: Discrete Mathematics Using Latin Squares. Wiley, New York (1998)
[Li] Li, W.: Modular curves and coding theory: a survey. In: Contemp. Math. vol. 518, 301–314. AMS, Providence (2010). Available: http://www.math.cts.nthu.edu.tw/download.php?filename=630_0d16a302.pdf&dir=publish&title=prep2010-11-001
[LN] Lidl, R., Niederreiter, H.: Finite Fields. Cambridge Univ. Press, Cambridge (1997)
[LvdG] Lint, J., van der Geer, G.: Introduction to Coding Theory and Algebraic Geometry. Birkhäuser, Boston (1988)
[Lo] Lorenzini, D.: An Invitation to Arithmetic Geometry. Grad. Studies in Math. AMS, Providence (1996)
[Los] Losonczi, L.: On reciprocal polynomials with zeros of modulus one. Math. Inequal. Appl. **9**, 286–298 (2006). http://mia.ele-math.com/09-29/~On-reciprocal-polynomials-with-zeros-of-modulus-one
[Lu] Luebeck, F.: Conway polynomials page. http://www.math.rwth-aachen.de:8001/~Frank.Luebeck/~data/ConwayPol/index.html
[MS] MacWilliams, F., Sloane, N.: The Theory of Error-Correcting Codes. North-Holland, Amsterdam (1977)
[Ma] Marden, M.: Geometry of Polynomials, American Mathematical Society, Providence (1970)
[MB] McEliece, R., Baumert, L.D.: Weights of irreducible cyclic codes. Inf. Control **20**, 158–175 (1972)
[MR] McEliece, R., Rumsey, H.: Euler products, cyclotomy, and coding. J. Number Theory **4**, 302–311 (1972)
[MN] MacKay, D.J.C., Radford, M.N.: Near Shannon limit performance of low density parity check codes. Electronics Letters, July (1996). Available: http://www.inference.phy.cam.ac.uk/mackay/abstracts/mncEL.html
[M] Mercer, I.D.: Unimodular roots of special Littlewood polynomials. Can. Math. Bull. **49**, 438–447 (2006)
[M1] Miller, R.: Graph automorphism computation (2007). http://www.rlmiller.org/talks/nauty.pdf
[M2] Miller, R.: Doubly even codes, June (2007). http://www.rlmiller.org/talks/June_Meeting.pdf
[Mo] Moreno, C.: Algebraic Curves over Finite Fields: Exponential Sums and Coding Theory. Cambridge Univ. Press, Cambridge (1994)
[N] Nakajima, S.: Galois module structure of cohomology groups for tamely ramified coverings of algebraic varieties. J. Number Theory **22**, 115–123 (1986)
[Nara] Narasimhan, R.: Complex Analysis of One Variable. Basel (1985)
[NRS] Nebe, G., Rains, E., Sloane, N.: Self-Dual Codes and Invariant Theory. Springer, Berlin (2006)
[NX] Nieddereiter, H., Xing, C.P.: Algebraic Geometry in Coding Theory and Cryptography. Princeton Univ. Press, Princeton (2009)
[Ni] Niven, I.: Coding theory applied to a problem of Ulam. Math. Mag. **61**, 275–281 (1988)
[O1] Ogg, A.: Elliptic curves with wild ramification. Am. J. Math. **89**, 1–21 (1967)

[O2] Ogg, A.: Modular Forms and Dirichlet series. Benjamin, Elmsford (1969). See also his paper in Modular Functions of One Variable, I, Proc. Internat. Summer School, Univ. Antwerp, Antwerp, 1972. Lecture Notes in Math., vol. 320, pp. 1–36. Springer, Berlin (1973)
[OEIS] The OEIS Foundation Inc.: http://oeisf.org/
[OM] O'Meara, T.: Introduction to Quadratic Forms. Springer, Berlin (2000)
[O] Ozeki, M.: On the notion of Jacobi polynomials for codes. Math. Proc. Camb. Philos. Soc. **121**, 15–30 (1997)
[PSvW] Pellikaan, R., Shen, B.-Z., Van Wee, G.J.M.: Which linear codes are algebraic-geometric? IEEE Trans. Inf. Theory **37**, 583–602 (1991). Available: http://www.win.tue.nl/math/dw/personalpages/ruudp/
[PS] Petersen, K., Sinclair, C.: Conjgate reciprocal polynomials with all roots on the unit circle. Preprint. Available: http://arxiv.org/abs/math/0511397
[Pl] Pless, V.: On the uniqueness of the Golay codes. J. Comb. Theory **5**, 215–228 (1968)
[P] Pretzel, O.: Codes and Algebraic Curves. Oxford Lecture Series, vol. 9. Clarendon, Oxford (1998)
[Ra] Rains, E.M.: Shadow bounds for self-dual codes. IEEE Trans. Inf. Theory **44**, 134–139 (1998)
[R1] Ritzenthaler, C.: Problèmes arithmétiques relatifs à certaines familles de courbes sur les corps finis. Thesis, Univ. Paris 7 (2003)
[R2] Ritzenthaler, C.: Action du groupe de Mathieu M_{11} sur la courbe modulaire $X(11)$ en caractéristique 3. Masters thesis, Univ. Paris 6 (1998)
[R3] Ritzenthaler, C.: Automorphismes des courbes modulaires $X(n)$ en caractristique p. Manuscr. Math. **109**, 49–62 (2002)
[Ro] Rovira, J.G.: Equations of hyperelliptic modular curves. Ann. Inst. Fourier, Grenoble **41**, 779–795 (1991)
[S] The SAGE Group: SAGE: Mathematical software, Version 4.6. http://www.sagemath.org/
[Scl] Schiznel, A.: Self-inversive polynomials with all zeros on the unit circle. Ramanujan J. **9**, 19–23 (2005)
[Sch] Schmidt, W.: Equations over Finite Fields: An Elementary Approach, 2nd edn. Kendrick Press (2004)
[Sc] Schoen, C.: On certain modular representations in the cohomology of algebraic curves. J. Algebra **135**, 1–18 (1990)
[Scf] Schoof, R.: Families of curves and weight distributions of codes. Bull. Am. Math. Soc. (NS) **32**, 171–183 (1995). http://arxiv.org/pdf/math.NT/9504222.pdf
[SvdV] Schoof, R., van der Vlugt, M.: Hecke operators and the weight distributions of certain codes. J. Comb. Theory, Ser. A **57**, 163–186 (1991). http://www.mat.uniroma2.it/~schoof/vdvhecke.pdf
[Se1] Serre, J.-P.: Quelques applications du théorème de densité de Chebotarev. Publ. Math. l'IHÉS **54**, 123–201 (1981). Available: http://www.numdam.org/item?id=PMIHES_1981__54__123_0
[Se2] Serre, J.-P.: Linear Representations of Finite Groups. Springer, Berlin (1977)
[Shim] Shimura, G.: Introduction to the Arithmetic Theory of Automorphic Functions. Iwanami Shoten and Princeton University Press, Princeton (1971)
[ShimM] Shimura, M.: Defining equations of modular curves. Tokyo J. Math. **18**, 443–456 (1995)
[Shok] Shokranian, S.: The Selberg-Arthur Trace Formula. Lecture Note Series, vol. 1503. Springer, Berlin (1992)
[SS] Shokranian, S., Shokrollahi, M.A.: Coding Theory and Bilinear Complexity. Scientific Series of the International Bureau, vol. 21. KFA Jülich (1994)
[Sh1] Shokrollahi, M.A.: Kapitel 9. In: Beitraege zur algebraischen Codierungs- und Komplexitaetstheorie mittels algebraischer Funkionenkoerper. Bayreuther mathematische Schriften, vol. 30, pp. 1–236 (1991)
[Sh2] Shokrollahi, M.A.: Stickelberger codes. Des. Codes Cryptogr. **9**, 203–213 (1990)
[Sin] Singleton, R.C.: Maximum distance q-nary codes. IEEE Trans. Inf. Theory **10**, 116–118 (1964)

[Sl] Sloane, N.J.A.: Self-dual codes and lattices. In: Relations Between Combinatorics and Other Parts of Mathematics. Proc. Symp. Pure Math., vol. 34, pp. 273–308. American Mathematical Society, Providence (1979)
[S73] Sloane, N.J.A.: Is there a $(72, 36)d = 16$ self-dual code. IEEE Trans. Inf. Theory **19**, 251 (1973)
[St1] Stepanov, S.: Codes on Algebraic Curves. Kluwer, New York (1999)
[St2] Stepanov, S.: Character sums and coding theory. In: Proceedings of the Third International Conference on Finite Fields and Applications, Glasgow, Scotland, pp. 355–378. Cambridge University Press, Cambridge (1996)
[Sti] Stichtenoth, H.: Algebraic Function Fields and Codes. Springer, Berlin (1993)
[T] Tarnanen, H.: An asymptotic lower bound for the character sums induced by the Legendre symbol. Bull. Lond. Math. Soc. **18**, 140–146 (1986)
[Tei] Teirlinck, L.: Nontrivial t-designs without repeated blocks exist for all t. Discrete Math. **65**, 301–311 (1987)
[Tho] Thompson, T.: From Error-Correcting Codes Through Sphere Packings to Simple Groups. Cambridge Univ. Press, Cambridge (2004)
[TV] Tsfasman, M.A., Vladut, S.G.: Algebraic-Geometric Codes, Mathematics and Its Applications. Kluwer Academic, Dordrecht (1991)
[TVN] Tsfasman, M.A., Vladut, S.G., Nogin, D.: Algebraic Geometric Codes: Basic Notions. Math. Surveys. AMS, Providence (2007)
[VSV] van der Geer, G., Schoof, R., van der Vlugt, M.: Weight formulas for ternary Melas codes. Math. Comput. **58**, 781–792 (1992)
[vdV] van der Vlugt, M.: Hasse-Davenport curves, Gauss sums, and weight distributions of irreducible cyclic codes. J. Number Theory **55**, 145–159 (1995)
[vL1] van Lint, J.: Introduction to Coding Theory, 3rd edn. Springer, Berlin (1999).
[vL2] van Lint, J.: Combinatorial designs constructed with or from coding theory. In: Longo, G. (ed.) Information Theory, New Trends and Open Problems. CISM Courses and Lectures, vol. 219, pp. 227–262. Springer, Wien (1975). Available: http://alexandria.tue.nl/repository/freearticles/593587.pdf
[vLW] van Lint, J., Wilson, R.M.: A Course in Combinatorics. Cambridge Univ. Press, Cambridge (1992)
[V] Velu, J.: Courbes elliptiques munies d'un sous-groupe $\mathbb{Z}/n\mathbb{Z} \times \mu_n$. Bull. Soc. Math. France, Memoire (1978)
[VVS] Viterbi, A.J., Viterbi, A.M., Sindhushayana, N.T.: Interleaved concatenated codes: New perspectives on approaching the Shannon limit. Proc. Natl. Acad. Sci. USA **94**, 9525–9531 (1997)
[V1] Voloch, F.: Asymptotics of the minimal distance of quadratic residue codes. Preprint. Available: http://www.ma.utexas.edu/users/voloch/preprint.html
[V2] Voloch, F.: Email communications (5-2006)
[V3] Voloch, F.: Computing the minimum distance of cyclic codes. Preprint. Available: http://www.ma.utexas.edu/users/voloch/preprint.html
[Wa] Wage, N.: Character sums and Ramsey properties of generalized Paley graphs. Integers **6**, A18 (2006). Available: http://integers-ejcnt.org/vol6.html
[War] Ward, H.N.: Quadratic residue codes and symplectic groups. J. Algebra **29**, 150–171 (1974)
[W] Weil, A.: On some exponential sums. Proc. Natl. Acad. Sci. **34**, 204–207 (1948)
[Z] Zhang, S.: On the nonexistence of extremal self-dual codes. Discrete Appl. Math. **91**, 277–286 (1999)

Index

D. Joyner, J.-L. Kim, *Selected Unsolved Problems in Coding Theory*,
Applied and Numerical Harmonic Analysis,
DOI 10.1007/978-0-8176-8256-9, © Springer Science+Business Media, LLC 2011

Applied and Numerical Harmonic Analysis

J.M. Cooper: *Introduction to Partial Differential Equations with MATLAB* (ISBN 978-0-8176-3967-9)

C.E. D'Attellis and E.M. Fernández-Berdaguer: *Wavelet Theory and Harmonic Analysis in Applied Sciences* (ISBN 978-0-8176-3953-2)

H.G. Feichtinger and T. Strohmer: *Gabor Analysis and Algorithms* (ISBN 978-0-8176-3959-4)

T.M. Peters, J.H.T. Bates, G.B. Pike, P. Munger, and J.C. Williams: *The Fourier Transform in Biomedical Engineering* (ISBN 978-0-8176-3941-9)

A.I. Saichev and W.A. Woyczyński: *Distributions in the Physical and Engineering Sciences* (ISBN 978-0-8176-3924-2)

R. Tolimieri and M. An: *Time-Frequency Representations* (ISBN 978-0-8176-3918-1)

G.T. Herman: *Geometry of Digital Spaces* (ISBN 978-0-8176-3897-9)

A. Procházka, J. Uhlíř, P.J.W. Rayner, and N.G. Kingsbury: *Signal Analysis and Prediction* (ISBN 978-0-8176-4042-2)

J. Ramanathan: *Methods of Applied Fourier Analysis* (ISBN 978-0-8176-3963-1)

A. Teolis: *Computational Signal Processing with Wavelets* (ISBN 978-0-8176-3909-9)

W.O. Bray and C.V. Stanojević: *Analysis of Divergence* (ISBN 978-0-8176-4058-3)

G.T Herman and A. Kuba: *Discrete Tomography* (ISBN 978-0-8176-4101-6)

J.J. Benedetto and P.J.S.G. Ferreira: *Modern Sampling Theory* (ISBN 978-0-8176-4023-1)

A. Abbate, C.M. DeCusatis, and P.K. Das: *Wavelets and Subbands* (ISBN 978-0-8176-4136-8)

L. Debnath: *Wavelet Transforms and Time-Frequency Signal Analysis* (ISBN 978-0-8176-4104-7)

K. Gröchenig: *Foundations of Time-Frequency Analysis* (ISBN 978-0-8176-4022-4)

D.F. Walnut: *An Introduction to Wavelet Analysis* (ISBN 978-0-8176-3962-4)

O. Bratteli and P. Jorgensen: *Wavelets through a Looking Glass* (ISBN 978-0-8176-4280-8)

H.G. Feichtinger and T. Strohmer: *Advances in Gabor Analysis* (ISBN 978-0-8176-4239-6)

O. Christensen: *An Introduction to Frames and Riesz Bases* (ISBN 978-0-8176-4295-2)

L. Debnath: *Wavelets and Signal Processing* (ISBN 978-0-8176-4235-8)

J. Davis: *Methods of Applied Mathematics with a MATLAB Overview* (ISBN 978-0-8176-4331-7)

G. Bi and Y. Zeng: *Transforms and Fast Algorithms for Signal Analysis and Representations* (ISBN 978-0-8176-4279-2)

J.J. Benedetto and A. Zayed: *Sampling, Wavelets, and Tomography* (ISBN 978-0-8176-4304-1)

E. Prestini: *The Evolution of Applied Harmonic Analysis* (ISBN 978-0-8176-4125-2)

O. Christensen and K.L. Christensen: *Approximation Theory* (ISBN 978-0-8176-3600-5)

L. Brandolini, L. Colzani, A. Iosevich, and G. Travaglini: *Fourier Analysis and Convexity* (ISBN 978-0-8176-3263-2)

W. Freeden and V. Michel: *Multiscale Potential Theory* (ISBN 978-0-8176-4105-4)

O. Calin and D.-C. Chang: *Geometric Mechanics on Riemannian Manifolds* (ISBN 978-0-8176-4354-6)

Applied and Numerical Harmonic Analysis (Cont'd)

J.A. Hogan and J.D. Lakey: *Time-Frequency and Time-Scale Methods* (ISBN 978-0-8176-4276-1)

C. Heil: *Harmonic Analysis and Applications* (ISBN 978-0-8176-3778-1)

K. Borre, D.M. Akos, N. Bertelsen, P. Rinder, and S.H. Jensen: *A Software-Defined GPS and Galileo Receiver* (ISBN 978-0-8176-4390-4)

T. Qian, V. Mang I, and Y. Xu: *Wavelet Analysis and Applications* (ISBN 978-3-7643-7777-9)

G.T. Herman and A. Kuba: *Advances in Discrete Tomography and Its Applications* (ISBN 978-0-8176-3614-2)

M.C. Fu, R.A. Jarrow, J.-Y. J. Yen, and R.J. Elliott: *Advances in Mathematical Finance* (ISBN 978-0-8176-4544-1)

O. Christensen: *Frames and Bases* (ISBN 978-0-8176-4677-6)

P.E.T. Jorgensen, K.D. Merrill, and J.A. Packer: *Representations, Wavelets, and Frames* (ISBN 978-0-8176-4682-0)

M. An, A.K. Brodzik, and R. Tolimieri: *Ideal Sequence Design in Time-Frequency Space* (ISBN 978-0-8176-4737-7)

B. Luong: *Fourier Analysis on Finite Abelian Groups* (ISBN 978-0-8176-4915-9)

S.G. Krantz: *Explorations in Harmonic Analysis* (ISBN 978-0-8176-4668-4)

G.S. Chirikjian: *Stochastic Models, Information Theory, and Lie Groups, Volume 1* (ISBN 978-0-8176-4802-2)

C. Cabrelli and J.L. Torrea: *Recent Developments in Real and Harmonic Analysis* (ISBN 978-0-8176-4531-1)

M.V. Wickerhauser: *Mathematics for Multimedia* (ISBN 978-0-8176-4879-4)

P. Massopust and B. Forster: *Four Short Courses on Harmonic Analysis* (ISBN 978-0-8176-4890-9)

O. Christensen: *Functions, Spaces, and Expansions* (ISBN 978-0-8176-4979-1)

J. Barral and S. Seuret: *Recent Developments in Fractals and Related Fields* (ISBN 978-0-8176-4887-9)

O. Calin, D. Chang, K. Furutani, and C. Iwasaki: *Heat Kernels for Elliptic and Sub-elliptic Operators* (ISBN 978-0-8176-4994-4)

C. Heil: *A Basis Theory Primer* (ISBN 978-0-8176-4686-8)

J.R. Klauder: *A Modern Approach to Functional Integration* (ISBN 978-0-8176-4790-2)

J. Cohen and A. Zayed: *Wavelets and Multiscale Analysis* (ISBN 978-0-8176-8094-7)

D. Joyner and J.-L. Kim: *Selected Unsolved Problems in Coding Theory* (ISBN 978-0-8176-8255-2)

For a fully up-to-date list of ANHA titles, visit http://www.springer.com/series/4968?detailsPage=titles *or* http://www.springerlink.com/content/t7k8lm/.