

Introduction to codes from a representation-theoretic perspective

David Joyner*

5-12-2005

This expository paper deals with some selected topics belonging to the intersection of the theory of error-correcting codes and the representation theory of finite groups. We shall see that codes which exhibit unusual symmetry often times turn out to be very interesting objects of study.

Notes of lectures given to undergraduate math majors at Harvey Mudd College, May 2005.

Contents

| | | |
|----------|---|-----------|
| 1 | Lecture 1: Codes and groups | 2 |
| 1.1 | Finite fields | 2 |
| 1.1.1 | Matrix representation | 2 |
| 1.1.2 | Conway polynomials | 3 |
| 1.2 | Linear codes: generalities | 4 |
| 1.3 | Linear codes: examples | 8 |
| 1.4 | Linear codes: automorphisms | 10 |
| 1.4.1 | Application to decoding | 11 |
| 2 | Lecture 2: Quadratic residue codes and other group codes | 12 |
| 2.1 | Cyclic codes revisited | 12 |
| 2.2 | Non-abelian group codes | 14 |
| 2.3 | QR codes | 14 |
| 2.3.1 | Fourier transforms on finite fields | 14 |
| 2.3.2 | Generalized quadratic residue codes | 16 |
| 2.3.3 | Extended quadratic residue codes | 17 |
| 3 | Lecture 3: Algebraic geometric codes for \mathbb{P}^1 | 18 |
| 3.1 | The projective line | 18 |
| 3.2 | Riemann-Roch spaces | 18 |
| 3.3 | The action of G on $L(D)$ | 19 |
| 3.4 | The codes | 21 |

*Math. Dept, USNA, wdj@usna.edu. I'd like to thank the Harvey Mudd Mathematics Department, especially Mike Orrison, for their hospitality.

1 Lecture 1: Codes and groups

Let \mathbb{F} denote a finite field. Since codes are vector spaces over finite fields, some very brief facts about finite fields will be recalled first.

1.1 Finite fields

We introduce some terminology and background about 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 $q = p^r$ 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. 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 $\mathbb{Z}/p\mathbb{Z}$. 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, there is only one field of order $q = p^r, r \geq 1$, denoted $GF(q)$. (Here “GF” stands for Galois field, named after the French mathematician E. Galois who died after a sword fight at the age of 23)

For any finite field \mathbb{F} , the multiplicative group of non-zero 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.

1.1.1 Matrix representation

Let \mathbb{E} denote a field extension of the finite field \mathbb{F} . Each element of \mathbb{E} may be represented as an invertible matrix with entries in \mathbb{F} . Here’s 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 non-zero 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

$$\rho : \mathbb{E}^\times \rightarrow \text{Aut}_{\mathbb{F}}(\mathbb{F}^m)$$

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 1 Taking $\mathbb{F} = GF(2)$ and $\mathbb{E} = GF(16)$ with defining polynomial $f(x) = x^4 + x^3 + 1$, we can represent the non-zero elements of $GF(16)$ as the following 15 matrices:

$$\begin{aligned}
& \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}, \\
& \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}, \\
& \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
\end{aligned}$$

Of course, matrix addition and multiplication corresponds to addition and multiplication of the corresponding field elements.

1.1.2 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 constructed by GAP and MAGMA.

We reproduce the definition on Frank Luebeck’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 $0, \dots, p-1$ of the cosets modulo p . We order these elements by $0 < 1 < 2 < \dots < p-1$. We introduce an ordering of the polynomials of degree r over $GF(p)$. Let $g(x) = g_r x^r + \dots + g_0$ and $h(x) = h_r x^r + \dots + h_0$ (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 $f_{p,m}(x)$.

These polynomials are not easy to compute but the fields $\mathbb{F}_1, \mathbb{F}_2, \dots$ constructed from a sequence

$$f_{p,r_1}, f_{p,r_2}, f_{p,r_3}, \dots \quad \text{with} \quad r_i | r_{i+1},$$

have “nice” embedding properties.

Sounds complicated but actually these fields are very easy to deal with using [Gap] or GUAVA, GAP’s error-correcting codes package [G1] (see the online GUAVA manual or [G2] for examples).

1.2 Linear codes: generalities

The theory of error-correcting codes was originated by Hamming in the late 1940's, a mathematician who worked for Bell Telephone. Some of his codes actually arose earlier in various isolated connections - for example, statistical design theory and in soccer betting(!). Hamming's motivation was to program a computer to correct "bugs" which arose in punch-card programs. The overall goal behind the theory of error-correcting codes is to reliably enable digital communication.

A (**linear error-correcting**) **code** C of length n over \mathbb{F} is a vector subspace of \mathbb{F}^n and its elements are called **codewords**. (When $\mathbb{F} = GF(2)$ it is called a **binary** code. These are the most important codes from the practical point of view.) Think of the following scenario: You are sending an n -vector of 0's and 1's (the codeword) across a noisy channel to your friend. Your friend gets a corrupted version (the received word differs from the codeword in a certain number of error positions). Depending on how the code C was constructed and the number of errors made, it is possible that the original codeword can be recovered. This raises the natural question: given C , how many errors can be corrected? Stay tuned...

A 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

$$0 \rightarrow \mathbb{F}^k \xrightarrow{G} \mathbb{F}^n \xrightarrow{H} \mathbb{F}^{n-k} \rightarrow 0. \quad (1)$$

("Short exact" means (1) the arrow G is injective, i.e., G is a full-rank $k \times n$ matrix, (2) the arrow H is surjective, and (3) $\text{image}(G) = \text{kernel}(H)$.) We identify C with the image of G . The function

$$\begin{aligned} G : \mathbb{F}^k &\rightarrow C, \\ \vec{m} &\longmapsto \vec{m}G, \end{aligned}$$

is called the **encoder**. Since the sequence (1) is exact, a vector $\vec{v} \in \mathbb{F}^n$ is a codeword if and only if $H(\vec{v}) = 0$. If \mathbb{F}^n is given the usual standard vector space basis then the matrix of G is a **generating matrix** of C and the matrix of H is a **check matrix** of C . In other words,

$$\begin{aligned} C &= \{\vec{c} \mid \vec{c} = \vec{m}G, \text{ some } \vec{m} \in \mathbb{F}^k\} \\ &= \{\vec{c} \in \mathbb{F}^n \mid H\vec{c} = \vec{0}\}. \end{aligned}$$

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 G is in **standard form**. By abuse of terminology, if this is the case then we say C is in **standard form**.

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 $\vec{e}_1, \dots, \vec{e}_k$ 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 .

Aside: For a "random" $k \times k$ matrix with **real** entries, the "probability" that its rank is k is of course 1. This is because "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 $GF(2)$ is invertible is

$$\lim_{k \rightarrow \infty} \frac{(2^k - 1)(2^k - 2) \dots (2^k - 2^{k-1})}{2^{k^2}} = \prod_{i=1}^{\infty} (1 - 2^{-i}) = 0.288\dots$$

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

The **Hamming metric** is the function

$$d : \mathbb{F}^n \times \mathbb{F}^n \rightarrow \mathbb{R},$$

$$d(\vec{v}, \vec{w}) = |\{i \mid v_i \neq w_i\}| = d(\vec{v} - \vec{w}, \vec{0}).$$

The **Hamming weight** of a vector is simply its distance from the origin:

$$\text{wt}(\vec{v}) = d(\vec{v}, \vec{0}).$$

Question: How many vectors belong to the “shell” of radius r about the origin $\vec{0} \in GF(q)^r$?

Answer: $\binom{n}{r} (q-1)^r$. Think about it! (Hint: “distance r ” means that there are exactly r non-zero coordinates. The binomial coefficient describes the number of ways to choose these r coordinates.)

The **minimum distance of C** is defined to be the number

$$d(C) = \min_{\vec{c} \neq \vec{0}} d(\vec{c}, \vec{0}).$$

(It is not hard to see that this is equal to the closest distance between any two distinct codewords in C .) An $[n, k]$ -code with minimum distance d is called an $[n, k, d]$ -**code**.

Cyclic construction Let G denote the cyclic group of order n . Let p denote a prime for which the finite field $\mathbb{F} = GF(p)$ contains a primitive n -th root of unity. Assume that G acts on \mathbb{F}^n by cyclically permuting the coordinates.

Pick a non-zero vector $\vec{a} \in \mathbb{F}^n$. Let $C \subset \mathbb{F}^n$ denote the vector space spanned by the cyclic permutations $g\vec{a}$, $g \in G$. In general, there seems to be no easy way to determine the minimum distance $d = d(C)$ from \vec{a} and G .

Let $G^* = \{\chi_1, \dots, \chi_n\}$ denote the dual group of G , where $\chi_i : G \rightarrow \mathbb{F}^\times$. (Since p is a prime for which \mathbb{F} contains all the n -th roots of unity, there are n such characters.)

The vector space C is G -invariant, by definition, so

$$C \cong m_1 \mathbb{F}[\chi_1] \oplus \dots \oplus m_n \mathbb{F}[\chi_n],$$

as G -modules, where $m_i \geq 0$ is the multiplicity of the i -th character. This is analogous to the fact that any “nice” complex-valued periodic function can be expanded in a Fourier series using powers of a complex exponential function.

As far as I know, there is no easy way to determine the minimum distance $d = d(C)$ from the characters χ_i 's and the “Fourier coefficients” m_i 's.

Example 2 Take $\mathbb{F} = GF(11)$, which contains all the 5-th roots of unity. Let $\alpha \in \mathbb{F}$ denote a 5-th root of unity (for example, take $\alpha = 4$). Let $\sigma : \mathbb{F}^5 \rightarrow \mathbb{F}^5$ denote the cyclic shift sending $(a, b, c, d, e) \mapsto (b, c, d, e, a)$ and let $G = \langle \sigma \rangle$. The \mathbb{F} -valued dual group of G , denoted G^* , is the set of functions $\chi_i = \chi^i$, where $\chi : G \rightarrow \mathbb{F}^\times$ is defined by

$$\chi(\sigma^i) = \alpha^i, \quad 0 \leq i \leq |G| - 1.$$

Let $\vec{c} = (1, 0, 2, 0, 3) \in \mathbb{F}^5$ and let G denote the 4×5 matrix whose rows are the cyclic shifts of \vec{c} . The row-reduced echelon form of G is

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 5 \\ 0 & 1 & 0 & 0 & 3 \\ 0 & 0 & 1 & 0 & 10 \\ 0 & 0 & 0 & 1 & 3 \end{pmatrix},$$

so the code C whose generator matrix is G is a cyclic $[5, 4, 2]$ -code over $GF(11)$.

How does this code decompose as a G -module? First, we must determine G -invariant basis vectors. To this end, let

$$\begin{aligned} \vec{w}_1 &= \vec{c} + \sigma(\vec{c}) + \sigma^2(\vec{c}) + \sigma^3(\vec{c}) + \sigma^4(\vec{c}), \\ \vec{w}_2 &= \vec{c} + \alpha\sigma(\vec{c}) + \alpha^2\sigma^2(\vec{c}) + \alpha^3\sigma^3(\vec{c}) + \alpha^4\sigma^4(\vec{c}), \\ \vec{w}_3 &= \vec{c} + \alpha^2\sigma(\vec{c}) + \alpha^4\sigma^2(\vec{c}) + \alpha^6\sigma^3(\vec{c}) + \alpha^8\sigma^4(\vec{c}), \\ \vec{w}_4 &= \vec{c} + \alpha^3\sigma(\vec{c}) + \alpha^6\sigma^2(\vec{c}) + \alpha^9\sigma^3(\vec{c}) + \alpha^{12}\sigma^4(\vec{c}). \end{aligned}$$

(Since $\alpha^5 = 1$ some of these exponents can be reduced if desired.) Note that $\sigma(\vec{w}_1) = \vec{w}_1$, $\sigma(\vec{w}_2) = \alpha^4\vec{w}_2$, $\sigma(\vec{w}_3) = \alpha^3\vec{w}_3$, and $\sigma(\vec{w}_4) = \alpha^2\vec{w}_4$. Therefore, these form a G -invariant basis and we have

$$C \cong \mathbb{F}[\chi_0] \oplus \mathbb{F}[\chi_1] \oplus \mathbb{F}[\chi_2] \oplus \mathbb{F}[\chi_3] \oplus \mathbb{F}[\chi_4].$$

In this case, every representation of G occurs in C , each with multiplicity one.

Lemma 1 (Singleton bound) Every linear $[n, k, d]$ code C satisfies

$$k + d \leq n + 1.$$

Note: this bound does not depend on the size of \mathbb{F} . A 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.

proof: Fix a basis of \mathbb{F}_q^n and write all the codewords in this basis. Delete the first $d - 1$ coordinates in each code word. Call this new code C' . Since C has minimum distance d , these codewords of C' are still distinct. There are therefore q^k of them. But there cannot be more than $q^{n-d+1} = |\mathbb{F}_q^{n-d+1}|$ of them. This gives the inequality. \square

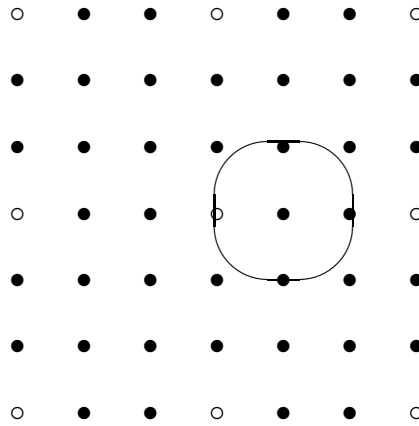
The **rate** of the code is $R = k/n$ - this measures how much information the code can transmit. The **relative minimum distance** of the code is $\delta = d/n$ - this is directly related to how many errors can be corrected.

Lemma 2 If $\vec{v} \in \mathbb{F}^n$ is arbitrary and $0 < r \leq \lfloor \frac{d-1}{2} \rfloor$ then the “ball” about \vec{v} with radius r ,

$$B_r(\vec{v}) = \{\vec{w} \in \mathbb{F}^n \mid d(\vec{v}, \vec{w}) \leq r\}$$

contains at most one codeword in C .

This follows easily from the fact that the Hamming metric is, in fact, a metric. Here is a picture of the idea.



Lemma 3 (*sphere-packing bound*) For any code $C \subset \mathbb{F}^n$, we have

$$|C| \sum_{i=0}^t \binom{n}{i} (q-1)^i \leq q^n,$$

where $t = \lfloor (d-1)/2 \rfloor$.

proof: For each codeword of C , construct a ball of radius t about it. These are non-intersecting, by definition of d and the previous lemma. Each such ball has

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

elements. The result follows from the fact that $\cup_{\vec{c} \in C} B_t(\vec{c}) \subset \mathbb{F}^n$ and $|\mathbb{F}^n| = q^n$. \square

Suppose (a) you sent $\vec{c} \in C$, (b) your friend received $\vec{v} \in \mathbb{F}^n$, (c) you know (or are very confident) that the number t of errors made is less than or equal to $\lfloor \frac{d-1}{2} \rfloor$. By the lemma above, the “ball” about \vec{v} of radius t contains a unique codeword. It must be \vec{c} , so your friend can recover what you sent (by searching through all the vectors in the ball and checking which one is in C) even though she/he only knows C and \vec{v} . This is called the **nearest neighbor decoding algorithm**:

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

Note “fail” is not returned unless $t > \lfloor \frac{d-1}{2} \rfloor$, by the above lemma.

Definition 1 We say that a linear C is **t -error correcting** if $|B_t(\vec{w}) \cap C| \leq 1$.

Note that $t \leq \lfloor \frac{d-1}{2} \rfloor$ if and only if $d \geq 2t + 1$.

The general goal in the theory is to optimize the following properties:

- the rate, $R = k/n$,
- the relative minimum distance, $\delta = d/n$,
- the speed at which a “good” encoder for the code can be implemented,
- the speed at which a “good” decoder for the code can be implemented.

There are (sometimes very technical) constraints on which these can be achieved, as we have seen with the Singleton bound and the sphere-packing bounds.

1.3 Linear codes: examples

We shall consider as an example one of the first codes constructed - one of an infinite family of codes called Hamming codes.

The Hamming [7, 4, 3] binary code: Let $\mathbb{F} = GF(2)$. The code C in this example has check matrix defined by

$$H = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} (\vec{H}_1, \vec{H}_2, \dots, \vec{H}_7) = \begin{pmatrix} \vec{h}_1 \\ \vec{h}_2 \\ \vec{h}_3 \end{pmatrix}$$

and generator matrix by

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} \vec{g}_1 \\ \vec{g}_2 \\ \vec{g}_3 \\ \vec{g}_4 \end{pmatrix}.$$

This code C is the $GF(2)$ -linear span of the rows $\vec{g}_1, \vec{g}_2, \vec{g}_3, \vec{g}_4$ of G .

Now try the following experiment: have a friend secretly pick $c_1, c_2, c_3, c_4 \in \mathbb{F}$, compute $\vec{c} = c_1\vec{g}_1 + c_2\vec{g}_2 + c_3\vec{g}_3 + c_4\vec{g}_4 \in C \subset GF(2)^7$, and tell you the 7 bits of \vec{c} , lying once. Suppose that they tell you $\vec{v} = (v_1, v_2, \dots, v_7)$. You can not only determine when they lied to you, but what the “secret” values $c_1, c_2, c_3, c_4 \in \mathbb{F}$ are.

Magic? No, but it is a neat trick. Here are 2 ways to do it.

Idea 1 (Syndromes): Compute the vector $\vec{s} = H\vec{v}$ (this vector is called a “syndrome”). Since the columns of H consist of all possible non-zero 3-tuples of 0’s and 1’s, if \vec{s} is non-zero then it must be one of the columns of H , say the i^{th} one. The vector \vec{c} is the same as \vec{v} but with the i^{th} bit flipped. Moreover, the first 4 coordinates of \vec{c} are the “secret” values $c_1, c_2, c_3, c_4 \in \mathbb{F}$.

Why does this work? First, if $\vec{c} \in C$ then, by definition of H , we must have $H\vec{c} = \vec{0}$. Let $\vec{H}_1, \vec{H}_2, \dots, \vec{H}_7$ denote the seven columns of H and let $\vec{e}_1, \vec{e}_2, \dots, \vec{e}_7$ denote the standard basis vectors of \mathbb{F}^7 . Note that

$$H\vec{v} = H(\vec{c} + \vec{e}_i) = H\vec{c} + H\vec{e}_i = \vec{0} + \vec{H}_i,$$

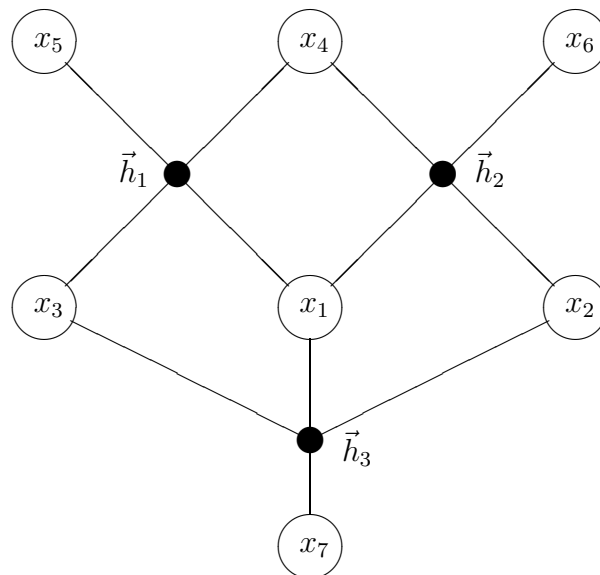
for each $1 \leq i \leq 7$. Since your friend lied in exactly one place, the column number of the syndrome is the place where the lie was made.

Natural questions:

- (a) How does this construction generalize? Does this decoder generalize?
- (b) Are there any “better” one error-correcting codes?
- (c) Are there analogous two error-correcting codes?

Hamming constructed a more general family of one error-correcting binary codes of length $n = 2^r - 1$, $r = 2, 3, 4, \dots$. The example above is the case $r = 3$. The decoder generalizes as well. There are no better one error-correcting codes. To some people, the “two error-correcting BCH codes” [MS] are the analogs of these Hamming codes. However, what family of linear two error-correcting codes have the best parameters is still an open question in general.

Idea 2 (Tanner graphs): Construct the bipartite graph Γ whose vertices V are labeled by the coordinates of the code (so $|V| = n$) and whose edges are labeled by the rows \vec{h}_i of a check matrix H of C (so $|E| = n - k$). In the above example, it is this following graph:



The check equations correspond to the solid black vertices, the coordinates of the codes to the labeled vertices, and the edges correspond to the terms occurring in the parity check equation. This is called a “Tanner graph” ([Ta], §B).

For each check vertex, add up the incident coordinate vertices (mod 2). The error position is determined by the following table:

| parity failure region(s) | error position |
|-----------------------------------|----------------|
| none | none |
| $\vec{h}_1, \vec{h}_2, \vec{h}_3$ | 1 |
| \vec{h}_2, \vec{h}_3 | 2 |
| \vec{h}_1, \vec{h}_3 | 3 |
| \vec{h}_1, \vec{h}_2 | 4 |
| \vec{h}_1 | 5 |
| \vec{h}_2 | 6 |
| \vec{h}_3 | 7 |

Natural questions:

- (a) Which bipartite graphs arise as a Tanner graph of a binary code?
- (b) The theory of bipartite graphs is extensive. Does the theory have useful coding-theoretic implications?
- (c) How does the Tanner graph depend on the choice of the check matrix H for C ?

See [Ta] for details and references in this direction. See also N. Sloane [S] for some unsolved problems associated with other graph-theoretic connections with coding theory.

1.4 Linear codes: automorphisms

What is an automorphism of a code? How do you construct a code with a “large” number of automorphisms? Can any finite group be realized as the automorphism group of a code? This section will address these questions.

To avoid some minor complications, we shall only deal with the simplest case of automorphisms of binary codes.

Let S_n denote the symmetric group on n letters. The **(permutation) automorphism group** of a code C of length n is simply the group

$$\text{Aut}(C) = \{\sigma \in S_n \mid (c_1, \dots, c_n) \in C \implies (c_{\sigma(1)}, \dots, c_{\sigma(n)}) \in C\}.$$

There is a more general definition of the automorphism group of a linear code over \mathbb{F} . In general, (a) the permutation automorphism group is always a subgroup of the full automorphism group, and (b) in the case of a binary linear code the two groups agree. For simplicity, here we only deal with the permutation automorphism group, which, for brevity, we simply call the automorphism group of C .

If C_1 and C_2 are two codes of length n and if there is a permutation $\sigma \in S_n$ for which $(c_1, \dots, c_n) \in C_1$ if and only if $(c_{\sigma(1)}, \dots, c_{\sigma(n)}) \in C_2$, then we say C_1 and C_2 are **permutation equivalent**. This will be written

$$C_1 \cong C_2.$$

It is a general fact that permutations preserve dimension and minimum distance: if $C_1 \cong C_2$ then $\dim(C_1) = \dim(C_2)$ and $d(C_1) = d(C_2)$.

Recall that the generator matrix G of an $[n, k, d]$ -code has rank k , and that the row-reduced echelon form of G , call it G' , has no rows equal to the zero vector. One immediate consequence of the row-reduction process is that one can permute the columns of G' , if necessary, to obtain a matrix of the 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. We have just verified the following result.

Lemma 4 *Any linear code is permutation equivalent to a code which is in standard form.*

Let C be a code, let $G = \text{Aut}(C) \subset S_n$ denote the (permutation) automorphism group, and let $\text{Aut}_{\mathbb{F}}(C)$ denote the automorphism group of C as an \mathbb{F} -vector space. We have a group homomorphism

$$\rho : G \rightarrow \text{Aut}_{\mathbb{F}}(C) \cong GL_k(\mathbb{F}),$$

defined as follows: an element in G is associated to the linear transformation which permutes the coordinates in the “obvious way”,

$$\sigma \longmapsto ((c_1, \dots, c_n) \in C \longmapsto (c_{\sigma^{-1}(1)}, \dots, c_{\sigma^{-1}(n)}) \in C).$$

Because of this, C is a representation space of G . Representation theory raises her head!

1.4.1 Application to decoding

We discuss permutation decoding - a decoding method which only works when you have a group action on the code.

Here is an extremely useful lemma.

Lemma 5 *Suppose $\vec{v} = \vec{c} + \vec{e}$, where $\vec{c} \in C$ and $\vec{e} \in \mathbb{F}^n$ is an “error vector” with Hamming weight $\text{wt}(\vec{e}) \leq t$. The information coordinates of \vec{v} are correct if and only if $\text{wt}(H\vec{v}) \leq t$.*

See [HP], §10.2.

Let G denote the permutation automorphism group of C . The **permutation decoding algorithm** is:

1. **Input:** A received vector $\vec{v} \in \mathbb{F}^n$.
Output: A codeword $\vec{c} \in C$ closest to \vec{v} .
2. For each $g \in G$, compute $\text{wt}(H(g\vec{v}))$ until one with $\text{wt}(H(g\vec{v})) \leq t$ is found (if none is found, the algorithm fails).
3. Extract the information symbols from $g\vec{v}$, and use G to compute codeword \vec{c}_g from them.
4. Return $g^{-1}\vec{c}_g = \text{Decode}(\vec{v})$.

This is implemented in GUAVA.

For example, if G acts transitively then permutation decoding will correct at least one error.

The key problem is to find a set of permutations in G which moves the non-zero positions in every possible error vector of weight $\leq t$ out of the information positions. (This set, called a **PD-set**, will be used in step 1 above instead of the entire set G .)

Natural questions:

- (a) Are there any interesting (useful and practical or “merely” mathematically beautiful) examples?
- (b) How does C decompose as a G -module?
- (c) Does its character contain interesting coding-theoretic information?
- (d) Is there a permutation list-decoder?

In the next lecture, question (a) is addressed. For (a)-(c), I refer to [JK1]. For a basic introduction to list decoding, see [Le].

2 Lecture 2: Quadratic residue codes and other group codes

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

2.1 Cyclic codes revisited

One of the simplest “group codes” is the family of cyclic groups, introduced in a very naive way in the last lecture. Here we use a more algebraic approach.

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

Definition 2 A linear code C of length n is a **cyclic code** if whenever $\mathbf{c} = (c_1, \dots, c_n)$ is a codeword then so is its cyclic shift $\mathbf{c}' = (c_2, \dots, c_n, c_1)$.

Example 3 Consider the binary code C with generator matrix

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

Clearly these four rows $\vec{g}_1, \vec{g}_2, \vec{g}_3, \vec{g}_4$ are obtained from the previous by a shift to the right. Also notes the shift of \vec{g}_4 to the right is equal to $\vec{g}_5 = \vec{g}_1 + \vec{g}_3 + \vec{g}_4$. The shift of \vec{g}_5 to the right is $\vec{g}_6 = \vec{g}_1 + \vec{g}_2 + \vec{g}_3$. And the shift of \vec{g}_6 is $\vec{g}_7 = \vec{g}_2 + \vec{g}_3 + \vec{g}_4$. The shift of \vec{g}_7 is \vec{g}_1 . Therefore, the linear code generated by G is invariant under shifts to the right. Therefore C is a cyclic code.

Cyclic codewords are conveniently represented as polynomials modulo $x^n - 1$. In fact, if $\vec{c} = (c_1, \dots, c_n)$ then let

$$c(x) = c_1 + c_2x + \dots + c_nx^{n-1}$$

denote the associated **codeword polynomial**. In this notation the cyclic shift $\vec{c}' = (c_2, \dots, c_n, c_1)$ of \vec{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 $x^n - 1$:

$$R_n = \mathbb{F}[x]/(x^n - 1). \tag{2}$$

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 6 *There is 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 fact GUAVA allows you to easily pass back and forth between codewords as vectors and codewords as polynomials.

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

Lemma 7 *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 non-zero element in I of smallest degree. Pick an arbitrary non-zero 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 $g(x) = s(x)$. \square

Definition 3 *Let C be a cyclic code of length n . Let I be the ideal corresponding to C by Lemma 6. We call $g(x)$ a **generator polynomial** of C if it is a generator of I .*

Example 4 We continue with Example 3. Let $g(x) = 1 + x^2 + x^3$. This is the codeword polynomial associated to the top row of the generator matrix. $g(x)$ is the generator polynomial of the cyclic code C . Note that $x^7 - 1 = (x + 1)(x^3 + x^2 + 1)(x^3 + x + 1)$.

2.2 Non-abelian group codes

The following construction generalizes the above example in an abstract way but will but 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 “ g -th 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

$$\lambda(g)(x^h) = x^{gh}, \quad g, h \in G.$$

(Note: $\lambda(g_1)\lambda(g_2)(x^h) = \lambda(g_1)x^{g_2h} = x^{g_1g_2h} = \lambda(g_1g_2)(x^h)$, for $g_1, g_2, h \in G$.) Therefore, λ defines an action of G on $\mathbb{F}[G]$ called the regular representation. Let the dimension of $\mathbb{F}[G]$ be denoted n (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 the G -orbit of a

$$G \cdot a = \{ \lambda(g)(a) \mid g \in G \}.$$

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 $G \cdot a$:

$$C = \text{Span}(\{ \lambda(g)(a) \mid g \in G \}) = \text{Span}(\left\{ \sum_{h \in G} c_{g^{-1}h} x^h \mid g \in G \right\}).$$

In this case, G acts on C the left by permuting coordinates via the left action of G on itself, so $G \subset \text{Aut}(C)$. More generally, one may take C to be any G -submodule of $\mathbb{F}[G]$.

2.3 QR 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].)

2.3.1 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 field of real numbers \mathbb{R} . (It is doubtful that Fourier had finite fields in mind in the early 1800's when he used Fourier series to solve the heat equation!) First, for the analog of e^{ixy} , we need to know how to construct the additive characters of \mathbb{F} .

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

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

By **quadratic reciprocity**, if $p > 2$ we have $\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$. If p, ℓ are both odd primes then we have $\left(\frac{\ell}{p}\right)\left(\frac{p}{\ell}\right) = (-1)^{\frac{(p-1)(\ell-1)}{4}}$. 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 $p \equiv 1 \pmod{8}$. If ξ is a p -th root of unity in a field containing F then every $w \in F(\xi)$ can be uniquely written as

$$w = w_0 + w_1\xi + w_2\xi^2 + \dots + w_{p-1}\xi^{p-1}, \quad w_i \in F.$$

Addition in $F(\xi)$ is as usual but multiplication is to be “performed $\pmod{\xi^p - 1}$ ”. We think of $F(\xi)$ as the analog of the field of complex numbers.

Define an additive character $\psi_1 : \mathbb{F} \rightarrow F(\xi)^\times$ 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.

Lemma 8 (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 $\psi : \mathbb{F} \rightarrow F(\xi)$ is any additive character of \mathbb{F} (i.e., satisfies $\psi(a_1 + a_2) = \psi(a_1)\psi(a_2)$, for all $a_1, a_2 \in \mathbb{F}$) 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 (which the interested reader who knows a little group theory might try on his/her own) can be found many books on group theory or finite fields.

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

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

Lemma 9 (*Fourier inversion*) If $f : \mathbb{F} \rightarrow F(\xi)$ 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, $|\mathbb{F}|^{-1}$ is to be regarded as an element of $F(\xi)$.)

This is a consequence of orthogonality.

2.3.2 Generalized quadratic residue codes

If “useful and practical” fought “mathematically beautiful” in a battle over the quadratic residue codes, “mathematically beautiful” would win. These codes seems to have reasonably fast encoders and decoders but lack good parameters¹. However, they have striking mathematical properties, especially as related to representation theory. We follow [MS], §16.4-16.5.

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 nonquadratic 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 $(\frac{\cdot}{p})$ defines a non-trivial 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 $\mathbb{F} = \{0, 1, \dots, p-1\}$. Now identify $GF(\ell)^p$ with the vector space of function values

$$\{(f(0), f(1), \dots, f(p-1)) \mid f : \mathbb{F} \rightarrow GF(\ell)\}.$$

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

$$C_Q(\mathbb{F}, F) = \{(f(0), f(1), \dots, f(p-1)) \mid FT_f(a) = 0, \quad \forall a \in Q\}.$$

There is an analogous code for the nonresidues:

$$C_N(\mathbb{F}, F) = \{(f(0), f(1), \dots, f(p-1)) \mid FT_f(a) = 0, \quad \forall a \in N\}.$$

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.

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

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

¹Although there are some extremely interesting but conjectural results in Bazzi-Mittel [BM] which use QR code-like constructions to construct related codes which seem to have very good parameters.

Lemma 10 *The parameters $[n, k, d]$ of the generalized quadratic residue codes satisfy*

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

Determining d for a quadratic residue code with p “large” is a very hard problem. For example, recently M. Grassl published some tables extending those in chapter 16 of [MS] of values $[n, k, d]$ for quadratic residue codes with $\ell = 2, 3$ and $p \leq 167$. Another apparently hard problem for these codes is to determine which coordinates the information bits lie in.

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

Definition 4 *If C is any $[n, k]$ -code over \mathbb{F} then the **dual code** C^\perp is a $[n, n - k]$ -code defined by the vector space of all n -vectors orthogonal every codeword:*

$$C^\perp = \{\vec{v} \in \mathbb{F}^n \mid \vec{v} \cdot \vec{c} = 0, \forall \vec{c} \in C\},$$

where

$$\vec{v} \cdot \vec{w} = v_1 w_1 + v_2 w_2 + \dots + v_n w_n \in \mathbb{F},$$

where $\vec{v} = (v_1, \dots, v_n)$, $\vec{w} = (w_1, \dots, w_n)$. A code satisfying $C^\perp = C$ is called **self-dual**.

Dual codes are often useful to have lying around. One nice property they have: a parity check matrix of C is a generating matrix for C^\perp .

Lemma 11

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

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

(This is proven in §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 themselves! (Such a code is sometimes called “self-orthogonal”.)

2.3.3 Extended quadratic residue codes

Define the **extended quadratic residue codes** by

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

$$\hat{C}_N(\mathbb{F}, F) = \{(c_1, \dots, c_p, c_\infty) \mid (c_1, \dots, c_p) \in C_N(\mathbb{F}, F), c_\infty = \alpha \sum_{i=1}^p c_i\},$$

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

See [MS], §16.5 for a proof of this and more details on how the permutation automorphism acts on the code (see also §6.6 of [HP]). 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 [W] for more of this fascinating story.

As was mentioned above, these codes seem to lack good parameters. However, work is still being done to improve estimates on the minimum distance d of these QR codes (see for example Voloch [V] and recent work of M. Grassl referenced there).

3 Lecture 3: Algebraic geometric codes for \mathbb{P}^1

Let $\mathbb{F} = GF(q)$ denote a finite field and let F denote an algebraic closure of \mathbb{F} .

In the early 1980's 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 distance can either be determined precisely or estimated in terms of some geometric parameters of the curve you started with. 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.

We must first build up some geometrical background before these codes can be introduced.

3.1 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 $F(\mathbb{P}^1)$. We shall, for reasons of space (pun intended), 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 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, $z \mapsto \frac{az+b}{cz+d}$, $\begin{pmatrix} a & b \\ c & d \end{pmatrix} \in GL(2, \mathbb{C})$. This action factors through $PGL(2, \mathbb{C})$ since scalar matrices act trivially. Similarly, $PGL(2, F)$ acts on $X = \mathbb{P}^1$. In fact, $\text{Aut}(X) = PGL(2, F)$.

3.2 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,

$$F(\mathbb{P}^1) = F(x).$$

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 non-zero. 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 non-zero is called the **support** of D , written $\text{supp}(D)$. The divisor of a rational function f is denoted $\text{div}(f)$. If f is, for example, a polynomial of degree n in x then $\text{div}(f) = P_1 + \dots + P_n - n\infty$ and $\text{supp}(\text{div}(f)) = \{P_1, \dots, P_n, \infty\}$, 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 is another divisor. The abelian group of all divisors is denoted $\text{Div}(\mathbb{P}^1)$.

Let $X = \mathbb{P}^1$ and let $F(X)$ denote the function field of X (the field of rational functions on X). If D is any divisor on X then the Riemann-Roch space $L(D)$ is a finite dimensional F -vector space given by

$$L(D) = L_X(D) = \{f \in F(X)^\times \mid \text{div}(f) + D \geq 0\} \cup \{0\},$$

where $\text{div}(f)$ 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 $\infty = [1 : 0] \in X$ 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 $\ell(D) = \deg(D) + 1$.

3.3 The action of G on $L(D)$

Let $X = \mathbb{P}^1/F$, so $\text{Aut}(X) = PGL(2, F)$, where F is algebraically closed.

The action of $\text{Aut}(X)$ on $F(X)$ is defined by

$$\begin{aligned} \rho : \text{Aut}(X) &\longrightarrow \text{Aut}(F(X)), \\ g &\longmapsto (f \longmapsto f^g) \end{aligned}$$

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, $\text{Aut}(X)$ also acts on the group $\text{Div}(X)$ of divisors of X , denoted $g(\sum_P d_P P) = \sum_P d_P g(P)$, for $g \in \text{Aut}(X)$, P a prime divisor, and $d_P \in \mathbb{Z}$. It is easy to show that $\text{div}(f^g) = g(\text{div}(f))$. Because of this, if $\text{div}(f) + D \geq 0$ then $\text{div}(f^g) + g(D) \geq 0$, for all $g \in \text{Aut}(X)$. In

particular, if the action of $G \subset \text{Aut}(X)$ on X leaves $D \in \text{Div}(X)$ stable then G also acts on $L(D)$. We denote this action by

$$\rho : G \rightarrow \text{Aut}(L(D)).$$

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 12 *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*

$$\{1, m_{P_i}(x)^k \mid 1 \leq k \leq a_i, 0 \leq i \leq s\}$$

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 1-dimensional vector space) be any non-zero element. Then*

$$\{m_P(x)^i q(x) \mid 0 \leq i \leq d\}$$

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.

In general, we have the following result.

Theorem 2 *Let $X, F, G \subset \text{Aut}(X) = \text{PGL}(2, F)$, and $D = \sum_{i=0}^s a_i P_i$ be a divisor as above. Let $\rho : G \rightarrow \text{Aut}(L(D))$ denote the associated representation. This acts trivially on the constants (if any) in $L(D)$; we denote this action by 1. Let $S = \text{supp}(D)$ and let*

$$S = S_1 \cup S_2 \cup \dots \cup S_m$$

be the decomposition of S into primitive G -sets.

(a) *If D is effective then*

$$\rho \cong 1 \oplus_{i=1}^m \rho_i,$$

where ρ_i is a representation on the subspace

$$V_i = \langle m_P(x)^{\ell_j} \mid 1 \leq \ell_j \leq a_j, P \in S_i \rangle,$$

satisfying $\dim(V_i) = \sum_{P_j \in S_i} a_j$, for $1 \leq i \leq m$. Here $\langle \dots \rangle$ denotes the vector space span.

(b) *If $\deg(D) > 0$ but D is not effective then ρ is a subrepresentation of $\rho : G \rightarrow \text{Aut}_F L(D')$, where D' is a G -equivariant effective divisor satisfying $D' \geq D$.*

proof: (a) Fix an i such that $1 \leq i \leq m$. Consider the subspace V_i of $L(D)$. Since G acts by permuting the points in S_i transitively, this action induces an action ρ_i on V_i . This action on V_i is irreducible since the action on S_i is transitive, by definition. Clearly $\oplus_{i=1}^m \rho_m$ is a subrepresentation of ρ . For dimension reasons, this subrepresentation must be all of ρ , modulo the constants (the trivial representation).

(b) Since D is not effective, we may write $D = D^+ - D^-$, where D^+ and D^- are non-zero effective divisors. The action of G must preserve D^+ and D^- . Since $L(D)$ is a G -submodule of $L(D^+)$, the claim follows. \square

3.4 The codes

Let D be a divisor in $X(\mathbb{F})$ stabilised by G whose support is contained in $X(\mathbb{F})$. Let $P_1, \dots, P_n \in X(\mathbb{F})$ be distinct points and $E = P_1 + \dots + P_n \in \text{Div}(X)$ be stabilized by G . This implies that G acts on the set $\text{supp}(E)$ by permutation. Assume $\text{supp}(D) \cap \text{supp}(E) = \emptyset$. Choose an \mathbb{F} -rational basis for $L(D)$ and let $L(D)_{\mathbb{F}}$ denote the corresponding vector space over \mathbb{F} . Let $C = C(D, E)$ denote the **algebraic geometric code**

$$C = \{(f(P_1), \dots, f(P_n)) \mid f \in L(D)_{\mathbb{F}}\}. \quad (3)$$

This is the image of $L(D)_{\mathbb{F}}$ under the evaluation map

$$\begin{aligned} \text{eval}_E : L(D) &\rightarrow F^n, \\ f &\longmapsto (f(P_1), \dots, f(P_n)). \end{aligned} \quad (4)$$

These are also called ‘‘classical Goppa codes’’. The group G acts on C by $g \in G$ sending

$$c = (f(P_1), \dots, f(P_n)) \in C \longmapsto c' = (f(g^{-1}(P_1)), \dots, f(g^{-1}(P_n))),$$

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 $c = (f'(P_1), \dots, f'(P_n)) \in C$, then we can easily verify $(f(g^{-1}(P_1)), \dots, f(g^{-1}(P_n))) = (f'(g^{-1}(P_1)), \dots, f'(g^{-1}(P_n)))$. (Indeed, G acts on the set $\text{supp}(E)$ by permutation.) This map $\phi(g)$ induces a homomorphism of G into the permutation automorphism group of the code $\text{Aut}(C)$, denoted

$$\phi : G \rightarrow \text{Aut}(C). \quad (5)$$

Let P be the permutation automorphism group of the code $C = C(D, E)$ defined in (3). In many cases it is known that the map $\phi : G \rightarrow P$ is an isomorphism (see [JK2], [We]). In any case, using (5), we regard C as a G -module. In particular, the (bijective) evaluation map $\text{eval}_E : L(D) \rightarrow C$ in (4) is G -equivariant. Since G acts (via ϕ) as a permutation on C , we have proven the following result.

Proposition 1 *Under the conditions above, the representation ρ of G on $L(D)$ is equivalent to a representation ρ' with property that, for all $g \in G$, $\rho'(g)$ is a permutation matrix.*

3.5 Memory application

If C is an linear code with non-trivial permutation group then this extra symmetry of the code may be useful in practice. In order to store the elements of C , we need only store one element in each G -orbit, so this symmetry can be used to more efficiently store codewords in memory on a computer.

Acknowledgement: Parts of these notes have been copied verbatim from joint work with my colleague W. Traves [JT] and with my students W. Irons [I] and J. McGowan [Mc].

References

- [Ba] A. Barg, UMCP web page, <http://www.enee.umd.edu/~abarg/>
- [BM] L. Bazzi and S. Mitter, *Some constructions of codes from group actions*, preprint, 2001.
- [Gap] The GAP Group, **GAP – Groups, Algorithms, and Programming, Version 4.4**; 2002, (<http://www.gap-system.org>).
- [G1] GUAVA home <http://cadigweb.ew.usna.edu/~wdj/gap/GUAVA/>
- [G2] GUAVA examples http://cadigweb.ew.usna.edu/~wdj/gap/GUAVA/GUAVA_examples.html
- [HP] W. C. Huffman and V. Pless, **Fundamentals of error-correcting codes**, Cambridge Univ. Press, 2003.
- [I] W. Irons, *A polynomial-time probabilistic algorithm for the minimum distance of an arbitrary linear non-binary error-correcting code*, <http://cadigweb.ew.usna.edu/~wdj/irons/>
- [JK1] D. Joyner and A. Ksir, *Modular representations on some Riemann-Roch spaces of modular curves $X(N)$* , in **Computational Aspects of Algebraic Curves** (Editor: T. Shaska), Lecture Notes in Computing, World Scientific, 2005.
- [JK2] ——— and ———, *Automorphism groups of some AG codes*, to appear (in PAMS?)
- [JT] ——— and W. Traves, “Representations of finite groups on Riemann-Roch spaces,” 2003 preprint, available at <http://front.math.ucdavis.edu/math.AG/0210408>
- [Lo] D. Lorenzini, **An invitation to arithmetic geometry**, Grad. Studies in Math, AMS, 1996.
- [Le] C. Lennon, *List-decoding of generalized Reed-Solomon codes Using Sudan’s algorithm*, <http://cadigweb.ew.usna.edu/~wdj/lennon/>
- [Lu] F. Luebeck, *Conway polynomials page*, <http://www.math.rwth-aachen.de:8001/~Frank.Luebeck/data/ConwayPol/index.html>

- [MS] F. MacWilliams and N. Sloane, **The theory of error-correcting codes**, North-Holland, 1977.
- [Mc] J. McGowan, *Implementing generalized Reed-Solomon codes and a cyclic code decoder in GUAVA*, <http://cadigweb.ew.usna.edu/~wdj/mcgowan/>
- [S] N. Sloane, *Unsolved Problems in graph theory arising from the study of codes*, in **Graph Theory Notes of New York** 18 (1989), pp. 11-20.
- [Ta] R. M. Tanner, *A transform theory for a class of group-invariant codes*, IEEE Trans. Infor. Theory 1988
- [vLM] J. van Lint, F. J. MacWilliams, "Generalized quadratic residue codes," Proc. IEEE Trans Info Theory 24(1978)730-737.
- [V] J. Voloch, *Computing the minimum distance of cyclic codes*, preprint available on the web-page
<http://www.ma.utexas.edu/users/voloch/preprint.html>
- [W] H. N. Ward, *Quadratic residue codes and symplectic groups*, J. Algebra 29(1974)150-171.
- [We] S. Wesemeyer, "On the automorphism group of various Goppa codes," *IEEE Trans. Info. Theory.*, 44(1998)630-643.