QR Codes

Henry Blanchette

Contents

0 Preface **3**

1 Introduction **3**

1.1 The Coding Problem . . . 3

1.2 Transmission Specifications . . . 3

2 Shannon's Theorem **4**

3 Linear Codes **6**

3.1 Linear Codes . . . 6

3.2 Hamming Codes . . . 7

3.3 Weight Enumerators . . . 8

4 The Binary Golay Code **8**

4.1 The Binary Golay Code . . . 8

5 Cyclic Codes **8**

5.1 Definitions . . . 8

5.2 Generator Matrix and Check Polynomial . . . 9

5.3 Zeros of a Cyclic Code . . . 10

5.4 Idempotent of a Cyclic Code . . . 10

5.5 Quadratic Residue (QR) Codes . . . 11

0 Preface

1 Introduction

1.1 The Coding Problem

Consider the scenario where a person, Alex, wants to send a message another person, Beth, via a noisy electrical channel. To facilitate such a transmission, a few pieces of equipment and processes are involved. First, Alex comes up with the message that he would like to transmit and writes it down in the form of an m-tuple,

$$\mathbf{a} = (a_m, \ldots, a_m).$$

Then, Alex uses a machine called a **encoder** that maps $\mathbf{a}$ to an n-tuple,

$$\mathbf{x} = (x_1, \ldots, x_n).$$

$\mathbf{x}$ is a **codeword -** one of some number of possible codewords in the encoder's image. Note that there must be at least as many codewords as there are possible original messages.

Next, x is transmitted to Beth as an electrical signal along a channel. During the transmission, some random noise $\mathbf{e}$ is added to the signal, where $\mathbf{e}$ is a n-tuple. The resulting signal that Beth recieves is $\mathbf{r} := \mathbf{x} + \mathbf{e}$.

In an attempt to correctly recover $\mathbf{a}$ from $\mathbf{r}$, Beth uses a machine called an **decoder**. The decoder calculates the most likely codeword $\mathbf{x}'$ that could have resulted in $\mathbf{r}$, and then outputs the message $\mathbf{a}'$ that corresponds to $\mathbf{x}'$ via inverse-encoding. If $\mathbf{r}$ is exactly a codeword, then $\mathbf{x}' = \mathbf{r}$. However, if $\mathbf{r}$ is not exactly a codeword, then the decoder finds the *closest* codeword to $\mathbf{r}$ in the space of the encoder's codomain (recall that are a subset of this codomain, the encoder's image).

The **coding problem** is the problem of devising an encoder/decoder pair that efficiently (in regards to some set of concerned features) and accurately faciliates transmissions like the one above. A construction of codewords of length n is referred to as a **code**, C.

1.2 Transmission Specifications

There is one possible "solution" to the coding problem that illustrates why specifying some more bounds on the transmission process is useful. Say there is a similar setup to the one in the previous section, and Alex wants to send Beth information about his coin-tossing prowess. After each toss, Alex sends the result to Beth in the form of a 0 for heads and 1 for tails. Alex tosses his coin at a speed of t tosses per minute. The channel connecting Alex and Beth is noisy such that there is a chance p that a bit is sent incorrectly, and a chance $q := 1 - p$ that bit is sent correctly. This channel is called a **binary symmetric channel**. Also, this channel only allows Alex to send $2t$ bits per minute and only during his coin-tossing session. When Alex gets a heads he transmits 0, and when he gets a tails he transmits 1. Alex decides to carry out his session for T minutes. At the end of the T minutes, Beth looks at the bits she received. She knows that a fraction p of them are incorrect, because of the channel's error rate. How could she reduce her decoding error lower than p?

Consider setup differing only in one aspect: there is no time constraint. Then instead of just sending one 0 or 1 for each toss, Alex can send N 0s or 1s for each toss. Then, Alice's decoder can decodes each section of N bits by taking the most common bit. Using this method, the probability of decoder error is

$$P_e(N) := \sum_{0 \le k \le N/2} \binom{N}{k} q^k p^{N-k}. \tag{1}$$

Furthermore,

$$\lim_{N \to \infty} P_e(N) = 0$$

so Alex and Beth can achieve arbitrarily accurate communication given enough time. The time constraint was an important obstacle after all!

2 Shannon's Theorem

The obviously unsatisfying aspect of the "solution" in section 1.2, other than the ignorance of a time constraint, is that it is extremely wasteful. There should have to be a good excuse for having to send a message any more than once. In turns out that, in fact, there are much better ways of achieving accuracy even within time and other constraints. Shannon's theorem states that, in the same situation as originally described in 1.2, Alex and Beth can still achieve arbitrarily small error probability.

Definition 2.0.1. Let C be a code with codewords of length n. Then the **information rate** of the code is

$$R := n^{-1} \log_2 |C|.$$

Definition 2.0.2. Let $\mathbf{x}, \mathbf{y} \in \mathbb{F}_2^n$. Then their **Hamming distance** is

$$d(\mathbf{x}, \mathbf{y}) := |\, \{i : x_i \neq y_i\} \,|$$

Suppose we have a binary symmetric channel with transmission-error probability p, and $q := 1-p$. Let $C = \{\mathbf{x}_i\}$ be a code of M words of length n, where each of the words are encoded to with equal probability. Suppose the decoder uses **maximum-likelihood** decoding i.e. the decoder decodes a received signal to the codeword that was most likely to be the original signal. Let P_i be the probability that the decoder is incorrect given that $\mathbf{x}_i$ is transmitted. So, the probability of an incorrect decoding is

$$P_C := M^{-1} \sum_{i=1}^{M} P_i \tag{2}$$

Finally, define

$$P^*(M, n, p) := \min \{P_C : C \text{ is a code with } M \text{ words of length } n \ \} \tag{3}$$

Theorem 2.0.1. *(Shannon's theorem)*

$$0 < R < 1 + p \log p + q \log q \implies \lim_{n \to \infty} P^*(M_n, n, p) = 0$$

where $M_n := 2^{[Rn]}$ *and all logarithms have base 2.*

Proof. Observe that the probability of an error pattern with w errors is $p^w q^{-w}$, which depends only on w. Denote the probability of receiving $\mathbf{y}$ given that $\mathbf{x}$ is transmitted by $P(y|x)$. Then also note that $P(\mathbf{y}|\mathbf{x}) = P(\mathbf{x}|\mathbf{y})$.

The number of errors in a received word is a random variable with expected value np and variance $np(1-p) = npq$. Let $\epsilon > 0$ and

$$b := \left(\frac{np(1-p)}{\epsilon/2} \right)^{1/2}.$$

Then by Chebyshev's inequality (TODO: cite 1.4.1), we have

$$P(w > np + b) \leq \frac{\epsilon}{2} \tag{4}$$

Let $\rho := \lfloor np + b \rfloor$. Then since $p < \frac{1}{2}$, ρ is less than $\frac{n}{2}$ when n is sufficiently large (TODO: show this). Define

$$B_\rho(\mathbf{x}) := \{\mathbf{y} : d(\mathbf{x}, \mathbf{y}) \leq \rho\} \tag{5}$$

which is the ball of radius ρ around $\mathbf{x}$. Then (TODO: cite Lemma 1.4.3) yields that

$$|B_\rho(\mathbf{x})| = \sum_{i\leq\rho}\binom{n}{i} < \frac{1}{2}\binom{n}{\rho} \leq \frac{n}{2}\frac{n^2}{\rho^\rho(n-\rho)^{n-\rho}} \tag{6}$$

We will use the following estimates:

$$\frac{\rho}{n}\log\frac{\rho}{n} = \frac{1}{n}\lfloor np+b\rfloor \log\frac{\lfloor np+b\rfloor}{n} = p\log p + O(n^{-1/2}) \tag{7}$$

$$\lim_{n\to\infty}\left(\left(1-\frac{\rho}{n}\right)\log\left(1-\frac{\rho}{n}\right)\right) = q\log q + O(n^{-1/2}) \tag{8}$$

Define

$$f : \mathbb{F}_2^n \times \mathbb{F}_2^n \to \mathbb{F}_2 \tag{9}$$

$$(\mathbf{u},\mathbf{v}) \mapsto \begin{cases} 0 & \text{if } d(\mathbf{u},\mathbf{v}) > \rho \\ 1 & \text{if } d(\mathbf{u},\mathbf{v}) \leq \rho \end{cases} \tag{10}$$

For $\mathbf{x}_i \in C$ and $\mathbf{y} \in \mathbb{F}_2^n$, define

$$\begin{aligned} g_i &: \mathbb{F}_2^n \to \mathbb{Z} \\ &\mathbf{y} \mapsto 1 - f(\mathbf{y},\mathbf{x}_i) + \sum_{j\neq i} f(\mathbf{y},\mathbf{x}_j) \end{aligned}$$

g_i is a function that counts the number of codewords other than $\mathbf{x}_i$ such that $d(\mathbf{x}_i,\mathbf{y}) \leq \rho$.

Now, choose M codewords $\mathbf{x}_1,\ldots,\mathbf{x}_M$ at random independently. Then the decoding algorithm is as follows. Suppose the decoder receives $\mathbf{y}$. If there is exactly one codeword $\mathbf{x}_i$ such that $d(\mathbf{x}_i,\mathbf{y}) \leq \rho$, i.e. $!\exists i : g_i(\mathbf{y}) = 0$, then decode $\mathbf{y}$ as $\mathbf{x}_i$. If there is not such $\mathbf{x}_i$, then the decoder has detected an error, and if it must decode anyway it outputs $\mathbf{x}_1$ as a default.

So P_i, the probability of error (as decidedd by the decoder algorithm), is such that

$$P_i \leq \sum_{\mathbf{y}\in\mathbb{F}_2^n} P(\mathbf{y}|\mathbf{x}_i)g_i(\mathbf{y}) = \sum_{\mathbf{y}} P(\mathbf{y}|\mathbf{x}_i)(1-f(\mathbf{y},\mathbf{x}_i)) + \sum_{\mathbf{y}}\sum_{j\neq i} P(\mathbf{y}|\mathbf{x}_j)f(\mathbf{y},\mathbf{x}_j)$$

where the right term is the probability that the received word $\mathbf{y}$ is not in $B_\rho(\mathbf{x}_i)$. By equation 4, $P_i \leq \frac{\epsilon}{2}$. Then,

$$P_C \leq \frac{\epsilon}{2} + M^{-1}\sum_{i=1}^{M}\sum_{\mathbf{y}}\sum_{j\neq i} P(\mathbf{y}|\mathbf{x}_i)f(\mathbf{y},\mathbf{x}_j).$$

Since $\mathbf{x}_1,\ldots,\mathbf{x}_M$ were chosen at random, we have

$$\begin{aligned} P^*(M,n,p) &\leq \frac{\epsilon}{2} + M^{-1}\sum_{i=1}^{M}\sum_{\mathbf{y}}\sum_{j\neq i}\mathcal{E}(P(\mathbf{y}|\mathbf{x}_i))\mathcal{E}(f(\mathbf{y},\mathbf{x}_j)) \\ &= \frac{\epsilon}{2} + M^{-1}\sum_{i=1}^{M}\sum_{\mathbf{y}}\sum_{j\neq i}\mathcal{E}(P(\mathbf{y}|\mathbf{x}_i))\cdot\frac{|B_\rho|}{2^n} \\ &= \frac{\epsilon}{2} + (M-1)2^{-n}|B_\rho|. \end{aligned}$$

Next, applying the estimates 7, we have

$$P^*(M,n,p) \leq n^{-1}\log(P^*(M,n,p) - \frac{\epsilon}{2}) \leq n^{-1}\log_M -(1+p\log p + q\log q) + O(\sqrt{n})$$

where $O(\sqrt{n})$ is a polynomial that is asymptotically equivalent to $\sqrt{n}$. Lastly we can substitute M_n for M, allowing the number of words, M in the code to depend on n, and use the restriction on R, $0 < R < 1 + p\log p + q\log q$, to get

$$n^{-1}\log(P^*(M_n,n,p) - \frac{\epsilon}{2}) < -\beta < 0$$

from the definition of R (definition 2.0.1), for

$$n > N := \frac{-\log \frac{\epsilon}{2}}{\beta}.$$

In other words,

$$P^*(M_n, n, p) < \frac{\epsilon}{2} + 2^{-\beta n}.$$

Thus

$$P^*(M_n, n, p) < \frac{\epsilon}{2} + 2^{-\beta n} < \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon.$$

□

This result was first published in C.E. Shannon's paper *Mathematical theory of communication* (1948), and is popularly recognized as origin of coding theory. The key concept that the theory illustrates is that **good codes** exist, where a good code is a code both usefully accurate yet more efficient than the unenightening code presented in section 1.2.

3 Linear Codes

Linear codes are the first step towards designing codes that have some algebraic structure. The symbols that a code uses are referred to as its **alphabet**. As shown in the previous section, binary codes have the alphabet $\mathbb{F}_2$, their name-sake. If the alphabet is taken to be some group Q, the the code is called a **group code**. For this section, we will use the $\mathbb{F}_q$ as the group for our group code, where $q = p^r$ for some prime p and some positive integer r. Then Q^n is is an n-dimensional vector space; denote Q^n by $\mathcal{R}^n$ or just $\mathcal{R}$. From here on, a **code** shall be defined to be a proper subset of $\mathcal{R}$.

Definition 3.0.1. The **minimum distance** of a nontrivial code C is

$$\min \{d(\mathbf{x}, \mathbf{y}) : \mathbf{x}, \mathbf{y} \in C, \mathbf{x} \neq \mathbf{y}\}$$

Definition 3.0.2. The **information rate** (or just **rate**) of a code C is

$$R := n^{-1} \log_q |C|$$

3.1 Linear Codes

Definition 3.1.1. A **linear code** C is a linear subspace of $\mathcal{R}$. Let k be the dimension of C. Then C is called an $[n, k]$ code. If C has a minimum distance between codewords, call it d. Then C is called an $[n, k, d]$ code.

Definition 3.1.2. A **generator matrix** G for a linear $[n, k]$ code C is a $k \times n$ matrix for which the rows are a basis of C. Observe that $C = \{\mathbf{a}G : \mathbf{a} \in \mathcal{R}\}$. G is in **standard form** (row-echelon form) if $G = (I\ P)$ where I is the $k \times k$ identity matrix. If G is in standard form, then the first k symbols of a codeword of C are called the **information symbols**, and the remaining symbols of the codeword are called the **parity check symbols**.

Definition 3.1.3. Let C be an $[n, k]$ code. Then C's **dual code**, $C^\perp$, is defined as

$$C^\perp := \{\mathbf{y} \in \mathcal{R}^n : \forall x \in C, \mathbf{x} \cdot \mathbf{y} = 0\}.$$

Note that $C^\perp$ is an $[n, n-k]$ code. If $\mathbf{y} \in C$, then $\forall x \in C : \mathbf{x} \cdot \mathbf{y} = 0$. The previous equation is called the **parity check equation** for C. Let $G = (I_k\ P)$ be the standard-formed generator matrix for C. Consider $H := (-P^\top \; I_{n-k})$ Since $GH^\top = 0$ (TODO: calculate this), every codeword $\mathbf{a}G$ has an inner product of 0 with each row of H, i.e.

$$\forall \mathbf{x} \in C : \mathbf{x}H^\top = \mathbf{0}, \tag{11}$$

which corresponds to a system of $n-k$ linear equations. In this way, H is a generator matrix for $C^\perp$. H is called the **parity check matrix** for C.

Definition 3.1.4. Let C be a linear code with parity check matrix H. Then for each $\mathbf{x} \in \mathcal{R}$, call $\mathbf{x}H^\top$ the **syndrome** of $\mathbf{x}$. Equation 11 demonstrated that C's codewords are characterized by the syndrome of $\mathbf{0}$.

C is a subgroup of $\mathcal{R}$ (TODO: prove this, from section 2.1). So we can partition $\mathcal{R}$ into cosets of C. For $\mathbf{x}, \mathbf{y} \in \mathcal{R}$ are in the same coset if and only if they have the same syndrome, i.e.

$$\mathbf{x}H^\top = \mathbf{y}H^\top \iff \mathbf{x} - \mathbf{y} \in C.$$

Therefor, if $\mathbf{r} = \mathbf{x} + \mathbf{e}$ is recieved by the decoder, where $\mathbf{x}$ is the original signal that the decoder *should* decode to and $\mathbf{e}$ is the added noise, then $\mathbf{r}$ and $\mathbf{e}$ have the same syndrome. In the maximum-likelihood decoding of $\mathbf{r}$, the decoder chooses an $\mathbf{e}$ of minimal weight such that $\mathbf{e}$ is in the same coset of $\mathbf{x}$, and then decodes $\mathbf{r}$ as $\mathbf{r} - e = \mathbf{x}$.

Definition 3.1.5. Let C be a code of length n over the alphabet $\mathbb{F}_q$. Then the **extended code** $\overline{C}$ is defined as

$$\overline{C} := \left\{ (c_1, \ldots, c_n, c_{n+1}) : (c_1, \ldots, c_n) \in C \wedge \sum_{i=1}^{n+1} c_i = 0 \right\}.$$

Let G be a generator and H be a parity check matrix for C. Then construct $\overline{G}$ by appending a column to G such that the sum of the columns of $\overline{G}$ is zero, making $\overline{G}$ indeed the parity check matrix for $\overline{C}$. Also construct $\overline{H}$ by

$$\overline{H} := \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ & & & & 0 \\ & & H & & 0 \\ & & & & \vdots \\ & & & & 0 \end{pmatrix} \tag{12}$$

Note that in the case that C is a binary code with odd minimum distance d, then $\overline{C}$ has minimum distance $d+1$ since the weights and distances for $\overline{C}$ must be even. (TODO: calcultate this)

3.2 Hamming Codes

Let G be the $k \times n$ generator matrix of an $[n, k]$ code C over $\mathbb{F}_q$. If any two columns of G are linearly independent (i.e. the columns when interpreted as vectors represent distinct points in $PG(k-1, q)$), then C is called a **projective code**. $C^\perp$ has G as its parity matrix. For $\mathbf{c} \in C^\perp$, if $\mathbf{e}$ is an error vector of weight 1, then the syndrome of $(\mathbf{c} + \mathbf{e})G^\top$ is a multiple of a column of G. In this way, $\mathbf{c} + \mathbf{e}$ uniquely determines one column of G, and so $C^\perp$ is a code that corrects at least one error.

Definition 3.2.1. Let $n := (q^k - 1)/(q-1)$. The $[n, n-k]$ **Hamming code** over $\mathbb{F}_q$ is a code for which the parity chekc matrix has columns which are pairwise linearly independent over $\mathbb{F}_q$ (i.e the columns are a maximal set of pairwise linearly independent vectors). The minimum distance of a Hamming code is 3. (TODO: prove why)

Example 3.2.1. The $[7, 4]$ binary Hamming code C has parity check matrix

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

(TODO: what to say about this, example 3.3.3)

3.3 Weight Enumerators

4 The Binary Golay Code

4.1 The Binary Golay Code

Consider the $[7,4]$ Hamming code H with the following parity check matrix:

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

H consists of $\mathbf{0}$, the seven cyclic shifts of (1 1 0 1 0 0 0), which is just $PG(2,2)$ (TODO: define this or throw out), and the complements of these 8 words (the complement of a word replaces 0s with 1s and visa versa). $\overline{H}$ Define H^* to be the code obtained by reversing the order of the symbols in the codewords of H. Consider the extended codes $\overline{H}, \overline{H^*}$, which are $[8,4]$ codes (TODO: calc this). These codes are such that $\overline{H} \cap \overline{H^*} = \{\mathbf{0}, \mathbf{1}\}$, since only these two words are in both codes forwards and backwards. Additionally both codes are self-dual and have minimum distance 4.

Next, define a code $\overline{C}$ with words of length 24 by

$$\overline{C} := \left\{(\mathbf{a}+\mathbf{x}, \mathbf{b}+\mathbf{x}, \mathbf{a}+\mathbf{b}+\mathbf{x}) : \mathbf{a}, \mathbf{b} \in \overline{H}, \mathbf{x} \in \overline{H^*}\right\}.$$

Observe that by letting $\mathbf{a}, \mathbf{b}$ range along a basis of $\overline{H}$ and $\mathbf{x}$ range along a basis of $\overline{H^*}$, $(\mathbf{a}, 0, \mathbf{a}), (0, \mathbf{b}, \mathbf{b}), (\mathbf{x}, \mathbf{x}, \mathbf{x})$ form a basis for $\overline{C}$. So $\overline{C}$ is a $[24,12]$ code. Furthermore, any two basis vectors of $\overline{C}$ are orthogonal and therefor $\overline{C}$ is self-dual as well. Since all basis vectors have weight divisible by 4 ((TODO: referenec to hamming codes?)), every word in $\overline{C}$ has weight divisible by 4.

Suppose that some $\mathbf{b} \in \overline{C}$ has $w(\mathbf{c}) < 8$. Since each of $\mathbf{x}+\mathbf{a}, \mathbf{b}+\mathbf{x}, \mathbf{a}+\mathbf{b}+\mathbf{x}$ have even weight, one of them must be $\mathbf{0}$. So, either $\mathbf{x} = \mathbf{0}$ or $\mathbf{x} = \mathbf{1}$. Without loss of generality, suppose $x = 0$. Then the vectors become $\mathbf{a}, \mathbf{b}, \mathbf{a}+\mathbf{b}$, which have weights 0, 4, or 8. Therefor $\mathbf{c} = \mathbf{0}$, and so $\overline{C}$ has minimum distance 8.

Next, construct the code C by removing the last coordinate of every word in $\overline{C}$. Then C is a $[23,12]$ code with minimum distance 7, since the last coordinate of each row of the generator matrix for H (which is also of the parity check matrix for H as defined at the beginning of this section) was a 1. The resulting code C is called the **binary Golay code**.

5 Cyclic Codes

5.1 Definitions

Definition 5.1.1. A linear code C is called **cyclic** if

$$(c_0, c_1, \dots, c_{n-1}) \in C \implies (c_{n-1}, c_0, \dots, c_{n-2}) \in C$$

An important fact going forward is the following isomorphism between $\mathbb{F}_q^n$ and a group of polynomials. The multiples of $x^n - 1$ form a principal ideal in the ring $\mathbb{F}[x]$. For the residue class (quotient) ring $\mathbb{F}_q/(x^n-1)$, the set of polynomials

$$\left\{\sum_{i=0}^{n-1} a_i x^i : a_i \in \mathbb{F}_q\right\}.$$

acts as a set of representatives for the equivalence classes. $\mathbb{F}_q^n$ is isomorphic to this quotient ring (with addition as its operation) via

$$(a_0, \dots, a_{n-1}) \leftrightarrow [a_0 x^0 + \cdots a_{n-1} x^{n-1}] \tag{13}$$

Additionally, in this polynomial ring, we can make use of polynomial multiplication. From now on, a codeword $\mathbf{c}$ may also be referred to as the polynomial $c(x) \in \mathbb{F}_q[x]/(x^n - 1)$ implicitly converting via equation 13.

Theorem 5.1.1. *A linear code C is cyclic if and only if C is an ideal in $\mathbb{F}_q[x]/(x^n - 1)$.*

Proof.

($\implies$) Suppose $c(x) = \sum c_i x^i$ is an ideal in $\mathbb{F}_q[x]/(x^n - 1)$. Then

$$xc(x) = \sum_{i=0}^{n-1} c_i x^{i+1} = c_{n-1}x^0 + \sum_{i=1}^{n-1} c_i x^i \mapsto (c_{n-1}, c_0, \ldots, c_{n-2}) \in C$$

and thus C is cyclic.

($\impliedby$) Suppose C is cyclic. Then $\forall c(x) \in C, xc(x) \in C$. Repeating this, we get $\forall i, x^i c(x) \in C$. Then since C is linear, this implies that $\forall a(x), a(x)c(x) \in C$, and hence C is an ideal.

□

From now on, we will only consider cyclic codes of length n over F_q with $(n, q) = (1)$.

Since $\mathbb{F}_q[x]/(x^n - 1)$ is a principal ideal domain (PID), every cyclic code C consists of the multiples of some polynomial $g(x)$, and $g(x)$ is the monic polynomial of least degree in the ring (a *monic* polynomial of degree d is one where the coefficient of x^d is 1). Call $g(x)$ the **generator polynomial** of the cyclic code C. Note that $g(x)$ divides $x^n - 1$ because if it did not, then $\gcd(g(x), x^n - 1)$ would be a polynomial of degree lower than that of $g(x)$.

Let $x^n - 1 = f_1(x) \cdots f_t(x)$ be a factoring into irreducibles. Since $(n, q) = (1)$, these factors must be different from each other. These irreducibles are all the possible options for generator polynomials of cyclic codes. For a chosen factor $f_i(x)$, the generated cyclic code is the set of multiple of $f_i(x)$ mod $x^n - 1$.

Definition 5.1.2. The cyclic code generated by $f_i(x)$ is called a **maximally cyclic code** and denoted by M_i^+; this is because $f_i(x)$ is a maximal idea in $\mathbb{F}_q[x]/(x^n - 1)$. The code generated by $(x^n - 1)/f_i(x)$ is called a **minimal cyclic code** and denoted M_i^-.

Observe that minimal cyclic codes are also *irredicible* cyclic codes because if there was a divisor $a(x)$ of $(x^n - 1)/f_i(x)$, then we would have (TODO: show contradiction).

5.2 Generator Matrix and Check Polynomial

Let $g(x)$ be the generator polynomial for a cyclic code C with codewords of length n. If $g(x)$ has degree $n - k$ the then the codewords $g(x), xg(x), \ldots, x^{k-1}g(x)$ form a basis for C. So C is an $[n, k]$ code. Writing $g(x)$ as $\sum_{i=0}^{n-k} g_i x^i$, we can construct a generator matrix for C as

$$G = \begin{pmatrix} g_0 & g_1 & \cdots & g_{n-k} & 0 & 0 & \cdots & 0 \\ 0 & g_0 & \cdots & g_{n-k-1} & g_{n-k} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & & & & \cdots & 0 \\ 0 & 0 & \cdots & & g_0 & g_1 & \cdots & g_{n-k} \end{pmatrix}$$

Using this form, we encode the information symbols of a codeword $(a_0, \cdots, a_{k-1})$ as $\mathbf{a}G$, which is just the polynomial

$$\left(\sum_{i=0}^{k-1} a_i x^i \right) g(x)$$

Recall that $g(x) \mid (x^2 - 1)$. So, there exists a polynomial $h(x) = \sum_{i=0}^{k} h_i x^i$ such that $g(x)h(x) = (x^n - 1)$ (in $\mathbb{F}_q[x]$). Then in $\mathbb{F}_q[x]/(x^n - 1)$, this yields $g(x)h(x) \equiv 0$, which is equivalent to the system of equations

$$\forall i \in \{0, \ldots, n-1\}, \sum_{j=0}^{n-k} g_j h_{i-j} = 0.$$

This yields that the parity check matrix for C is

$$\begin{pmatrix} 0 & 0 & \cdots & 0 & h_k & & \cdots & h_1 & h_0 \\ 0 & 0 & \cdots & h_k & \cdots & & h_1 & h_0 & 0 \\ \vdots & & & & & & & & \vdots \\ h_k & & \cdots & & h_1 & h_0 & 0 & \cdots & 0 \end{pmatrix}$$

and correspondingly $h(x)$ is the **check polynomial** of C. In this way, the code C is the set of codes $c(x)$ such that $c(x)h(x) = 0$.

Interestingly, the code generated by $h(x)$ is equivalent to the dual of the code code generated by $g(x)$, namely C. Recall that the dual $C^{\perp}$ is the code obtained by reversing each of the codewords of C. Nicely this results in the dual of a maximal cyclic code M_i^+ as being the minimal cyclic code M_i^-.

5.3 Zeros of a Cyclic Code

Theorem 5.3.1. *Let $n := (q^m - 1)/(q - 1)$, β be a primitive nth root of unity in $\mathbb{F}_q^m$, and $(m, q-1) = (1)$. Then the cyclic code*

$$C := \{c(x) : c(\beta) = 0\}$$

is equivalent to the $[n, n-m]$ Hamming code over $\mathbb{F}_q$.

Proof. Note that

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

Then we have that $(n, q-1) = (m, q-1)$. Therefor $\beta^{i(q-1)} \neq 1$ for $i = 1, \cdots, n-1$, i.e. $\beta^i \notin \mathbb{F}_q$. This implies that, since the columns in the parity check matrix H are the representations of $\beta^0, \ldots, \beta^{n-1}$ vectors in $\mathbb{F}_q^m$, these columns are linearly independent. Thus H is the parity check matrix of an $[n, n-m]$ Hamming code. □

5.4 Idempotent of a Cyclic Code

Theorem 5.4.1. *Let C be a cyclic code. Then there is a unique codeword $c(x)$ which is an identity element for C.*

Proof. Let $g(x)$ be the generator polynomial and $h(x)$ be the check polynomial of C. Recall from section 5.2 that $g(x)h(x) = (x^n - 1)$. Since $x^n - 1$ has no multiple zeros we have $(g(x), h(x)) = (1)$, and per usual we get polynomials $a(x), b(x)$ such that

$$g(x)a(x) + h(x)b(x) = 1.$$

Define

$$c(x) := a(x)g(x) = 1 - b(x)h(x).$$

$c(x)$ is a codeword in C because C is generated by $g(x)$. Let $p(x)g(x)$ be a codeword in C. Then

$$\begin{aligned} c(x)p(x)g(x) &= p(x)g(x) - b(x)h(x)p(x)g(x) \\ &\equiv p(x)g(x) \bmod (x^n - 1) \end{aligned}$$

since $b(x)h(x)p(x)g(x) = b(x)p(x)0 \bmod (x^n - 1)$. Hence $c(x)p(x)g(x) = p(x)g(x)$ implies that $c(x)$ is an identity element for C and is unique.

□

5.5 Quadratic Residue (QR) Codes

In this section, consider only codes with word length $n > 2$ prime. Additionally the alphabet, $\mathbb{F}_q$, of the code must satisfy the following: q is a quadratic residue mod n i.e. $q^{(n-1)/2} \equiv 1 \bmod n$. Let α denote a primitive nth root of unity in an extension field of $\mathbb{F}_q$. (TODO: what is this?) Define

$$\begin{aligned}
R_0 &:= \left\{ i^2 \bmod n : i \in \mathbb{F}_n \wedge i \neq 0 \right\}, \text{ the quadratic residues in } \mathbb{F}_n \quad &(14)\\
R_1 &:= F_n^* \setminus R_0, \text{ the non-residues in } \mathbb{F}_n \quad &(15)\\
g_0(x) &:= \prod_{r \in R_0} (x - \alpha^r) \quad &(16)\\
g_1(x) &:= \prod_{r \in R_1} (x - \alpha^r) \quad &(17)
\end{aligned}$$

Since q is a quadratic residues, $q \in R_0$, and thus the polynomials $g_0(x), g_1(x)$ have coefficients in $\mathbb{F}_q$ (q prime implies that $\mathbb{Z}/q\mathbb{Z}$ is a field). Furthermore,

$$x^n - 1 = (x-1)g_0(x)g_1(x) \tag{18}$$

(TODO: prove or cite)

Definition 5.5.1. The cyclic codes of length n over $\mathbb{F}_q$ with generators $g_0(x)$ and $(x-1)g_0(x)$ are both called **quadratic residue codes** (a.k.a QR codes).

In the case of binary codes, the condition that q is a quadratic residue mod n is equivalent to the condition that $n \equiv \pm 1 \bmod 8$. Let π_j be the permutation of the positions of codewords given by $i \mapsto ij \bmod n$. π_j maps the code with generator $g_0(x)$ into itself if $j \in R_0$ and maps the code with generator $g_1(x)$ into itself if $j \in R_1$. (TODO: proof or at least explanation?) This shows that all codes with generator $g_0(x)$ are equivalent, and respectively for codes with generator $g_1(x)$.

In the case that $n \equiv -1 \bmod 4$, then $-1 \in R_1$, and the transformation $x \mapsto x^{-1}$ maps a codeword of the code with generator $g_0(x)$ to into a codeword of the code with generator $g_1(x)$. (TODO: significance of this?)

Theorem 5.5.1. *Let* $\mathbf{c} = c(x)$ *be a codeword in the QR code with generator* $g_0(x)$ *such that* $c(1) \neq 0$. *Let* $d = w(\mathbf{c})$. *Then*

(i) $d^2 \geq n$.

(ii) $d \equiv -1 \textit{ mod } 4 \implies d^2 - d + 1 \geq n$.

(iii) $n \equiv -1 \textit{ mod } 8 \wedge q = 2 \implies d \equiv 3 \textit{ mod } 4$.

Proof.

(i) $c(x) \nmid (x-1)$ because $c(1) \neq 0 \implies c(x) \neq (x-1)$ and $(x-1)$ is irreducible. We can transform $c(x)$ into a polynomial $\hat{c}(x)$ which is divisible by $g_1(x)$ but still not divisible by $(x-1)$ via π_j for j. This implies that $c(x)\hat{c}(x)$ is a multiple of $\sum_{i=1}^{n-1} x^i$ because (TODO: how exactly?). Since $w(\mathbf{c}) = d$, $c(x)\hat{c}(x)$ has at most d^2 nonzero coefficients, and thus $d^2 \leq n$ where n is the length of the codeword.

(ii) Let $j = -1$ in the proof for (i). Then $\hat{c}(x)$ is the reverse of $c(x)$, and they overlap in at most d positions. Thus $c(x)\hat{c}(x)$ has at most $d^2 - d + 1$ nonzero coefficients. (TODO: how exactly?)

(iii) Write $c(x)$ as $\sim_{i=1}^{d} x^{l_i}$ and $\hat{c}(x) = \sum_{i=1}^{d} x^{-l_i}$. Note that for any indices i, j, k, l we have $l_i - l_j = l_k - l_l \implies l_j - l_i = l_l - l_k$. Then the products resulting in $c(x)\hat{c}(x)$ must cancle, if they do, in batches of fours. Therefor, we further deduce that $n = d^2 - d + 1 - 4a$ for some $a \geq 0$.

□

Theorem 5.5.2. *For a suitable choice of the primitive element α of $\mathbb{F}_q$, the polynomial*

$$\theta(x) := \sum_{i \in R_0} x^r$$

is the idempotent of the binary QR code with generator $(x-1)g_0(x)$ if $n \equiv 1$ mod 4 and is the idempotent of the QR code with generator $g_0(x)$ if $n \equiv -1$ mod 8.

Proof. θ is an idempotent polynomial (TODO: prove this). Therefor $\{\theta(\alpha)\}^2 = \theta(\alpha)$, and so it must be that $\theta(\alpha) = 0$ or $\theta(\alpha) = 1$. In the same way, $\theta(\alpha^i) = \theta(\alpha)$ if $i \in R_0$ and

$$\theta(\alpha^i) + \theta(\alpha) = 1$$

if $i \in R_1$. It is impossible for all possible α to satisfy $\theta(1) = 1$ (TODO: this is supposed ot be easy to realize), so the suitable choice for α is such that $\theta(\alpha) = 0$. This choice yields that $\theta(\alpha^i) = 0$ if $i \in R_0$ and $\theta(\alpha^i) = 1$ if $i \in R_1$. So, $\theta(\alpha^0) = (n-1)/2$ (TODO: why exactly?) □

Now, construct a matrix C (called a circulant) by taking the word θ as the first row and all cyclic shifts of it as the other rows. Let $\mathbf{c} := (0 \ \cdots \ 0)$ if $n \equiv 1$ mod 8, or $\mathbf{c} := (1 \ \cdots \ 1)$ if $n \equiv -1$ mod 8. Then defined

$$G := \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \mathbf{c}^\top & & C & \end{pmatrix} \tag{19}$$

Theorem 5.5.2 yields that the rows of G generate the extended binary QR code of length $n+1$. Now, number the coordinate places of the codewords of this code with the points of the projective line of order n: $\infty, 0, \ldots, n-1$. The parity check is the first coordinate, the ∞th place. The projective special linear group $PSL(2, n)$ consists of all transformations $x \mapsto (ax+b)/(cx+d)$ where $a, b, c, d \in \mathbb{F}_n$ and $ad - bc = 1$. This group is generated by the transformations

$$S(x) := x + 1 \qquad\qquad T(x) := -x^{-1}$$

Considering the usual algebraic treatment of ∞, S is a cyclic shift to the right for all positions other than ∞ and leaves the ∞th invariant. So, by definition 5.5.1, S leaves the extended code invariant. The effect of T is the mapping of a row of G into a linear combination of at most three rows of G ((TODO: reference citation [42])). Altogether, both of S, T leave the extended QR code invariant. This fact proves the following theorem.

Theorem 5.5.3. *The automorphism group of the extended binary QR code of length $n+1$ contains $PSL(2, n)$.*

Example 5.5.1. Let $q = 2, n = 7$. Then

$$x^7 - 1 = (x-1)(x^3 - x + 1)(x^3 + x^2 + 1).$$

Take the generator as $g_0(x)$. By theorem 5.5.2, we must have $x + x^2 + x^4$ as also a generator. Thus $g_0(x) = 1 + x + x^3$. This code is the (perfect) [7, 4] Hamming code.