Notes on Classical & Quantum Error-Correcting Codes

February 18, 2018

P. Ramond

Lecture 2: Classical Error Correction

Much of the following is borrowed from Thomas M. Thompson's beautiful little book, *From Error-Correcting Codes Through Sphere Packings to Simple Groups*", as well as Raymond Hill's *"A First Course in Coding Theory"*.

Complete treatments of classical error-correcting codes can be found in

Error-Correcting Codes W Wesley Peterson and E. J. Weldon, MIT Press (1961,1972)

Fundamentals of Error-Correcting Codes by Cary Huffman and Vera Pless, Cambridge University Press (2003).

1 Hamming

Consider 2^n words described by a string of n binary entries $(x_1, x_2, \ldots, x_n)$ of length n, where $x_i = 0, 1$.

In the transmission of a string (message) errors can occur as flips from one to zero and vice-versa in any of its entries. In order to guard against such flip errors, the recipient needs to be able to determine,

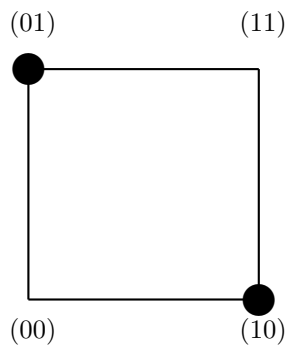
- if errors have occurred,
- and the locations of the errors in the length-n word entries.

A good error-correcting code is one for which both detection and location of flip errors are possible. Codes differ in terms of the amount of redundancy and the ease and speed of encoding and decoding.

Crude redundancy is achieved by duplicating a message, but this is slow and costly. A much better way is to include a message (codeword) with 2^k entries into a strings of length n (words). The remaining strings of length $r = (n - k)$ are check digits needed to diagnose and correct for errors.

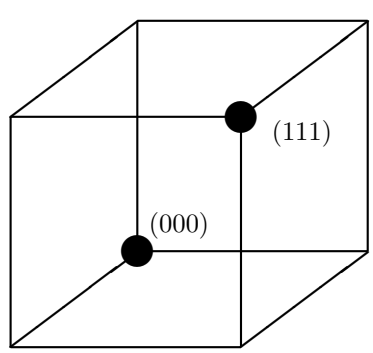
Illustrate these requirements with two simple examples,

The two-bit case consists of four states, at the four vertices of a square:



Place the codewords (10), (01) at the opposite vertices of the square. A single flip error will shift either to the other two vertices, and a double flip on either codeword will map it to itself or to the other codeword. The single error is detected when the received message is either (11) or (00), but there is an ambiguity in tracing the single flip to a codeword.

One can do better with three bits: Hamming places the eight words on the vertices of a cube:



He locates two codewords at (000) and (111), the opposite vertices of the cube. A single flip on either codeword moves along three directions, and do not overlap with the single flips of the other codeword. Now single flip errors are unambiguously identified. and it is double flip errors that are ambiguous. It gets worse as triple flips move one codeword to the other! This led Hamming to consider all words as vertices of hypercubes, and to some useful definitions:

Hamming weight: the number of ones in a word.

Hamming distance: the number of flips between two words.

Hamming sphere: the "sphere" centered at the codeword with radius equal to the number of flips.

In the 3-bit case, each Hamming sphere contains three vertices (words) with radius one. Since their sum exhausts the number of words it is a perfect code.

Hamming's Square Code (1947-49) is an early example of a single error detection classical code which relies on the binary form of the decimal digits. It contains 2^4 words expressed binary form: "0" = (0000), "1" = (0001), $\cdots$, "15" = (1111).

To transmit the digit nine = (1001), Hamming splits it in two parts 10 and 01 arranged in a square array. A third column and a third row are the (mod 2) sums of the horizontal and vertical codeword entries, resulting in a (3×3) array,

$$\begin{array}{cc|c} 1 & 0 & 1 \\ 0 & 1 & 1 \\ \hline 1 & 1 & 0 \end{array}$$

The transmitted message consists of nine binaries read horizontally from the array, so the digit "nine" is $\mathbf{1}\,\mathbf{0}\,1\,\mathbf{0}\,\mathbf{1}\,1\,1\,1\,0$, where the boldface symbols contain the information. Suppose a single codeword error occurs in transmission,

$$(\mathbf{1}\,\mathbf{0}\,1\,\mathbf{0}\,\mathbf{1}\,1\,1\,1\,0) \quad \Longrightarrow \quad (\mathbf{1}\,\mathbf{0}\,1\,\mathbf{1}\,\mathbf{1}\,1\,1\,1\,0).$$

This error is detected and located because the first column and second row parity sums do not check, which means that the 21 entry of the codeword has flipped. Should a flip occur in any of the five parity checks, it can be located in a similar way. This square code generalizes to codewords of length $n = (s+1)^2$, with s^2 message entries and $2s+1$ check digits.

Hamming's $(7,4)$ Code

One easily sees that the ninth binary entry is redundant, but can one go any further by lowering the number of check digits from 4 to 3? Note that 2^r check digits should be enough to identify n single flip errors and the correct codeword, since

$$2^r \geq n + 1,$$

and the bound is saturated for $n = 7$ and $r = 3$ check digits. Hamming does invent a single-error correcting code of length seven with three check digits by computing the check digits according to the decimal values of the codewords.

Insert the three check bits in the first (2^0), second (2^1), and fourth (2^2) positions of the length-7 words. The codewords are entered in binary form into the remaining third,fifth, sixth, and seventh positions.

Split the decimals into four groups according to their binary representation:

- decimals with a one in the first place (from the right) : 1, 3, 5, 7,

- decimals with a one in the second place (from the right) : 2, 3, 6, 7,

- decimals with a one in the third place (from the right) : 4, 5, 6, 7.

The values of the three check bits are computed using the following algorithm:

- The first check bit as the sum of the code bits in the 1st, 3rd, 5th, 7th positions,
- The second check bit as the sum of the code bits in the 2nd, 3rd, 6th, 7th positions ,
- The third check bit as the sum of the code bits in the 4th, 5th, 6th, 7th positions.

To these three sequences correspond the (3×7) matrix,

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

As an example, the decimal message $10 \sim (1\,0\,1\,0)$ lies in the message $(\mathfrak{A}\,\mathfrak{B}\,1\,\mathfrak{C}\,0\,1\,0)$. The parity checks are computed as the sum of the three code bits in the above sequences,

$$\mathfrak{A} = 1 + 0 + 0 = 1, \qquad \mathfrak{B} = 1 + 1 + 0 = 0, \qquad \mathfrak{C} = 0 + 1 + 0 = 1,$$

so that the message "10" is sent as the encoded codeword $(1\,0\,1\,1\,0\,1\,0)$.

The recipient of the message computes the *Error Syndrome*, the value of the three binary entries according to the above three sequences.

If the received message has no error, the three parity checks give $(0\,0\,0)$. On the other hand, suppose an error has occurred in the sixth position,

$$(1\,0\,1\,1\,0\,1\,0) \quad \Longrightarrow \quad (1\,0\,1\,1\,0\,\mathbf{0}\,0).$$

Compute the three check bits,

$$1+1+0+0=0, \qquad 0+1+0+0=1, \qquad 1+0+0+0=1,$$

which yields in reverse order the binary (1 1 0) form of the decimal "6": one concludes that the flip error was in the 6th position. This is a single error-correcting code.

Each of the 2^4 codewords is at the center of a single-error Hamming sphere with $7+1=2^3$ points. The Hamming spheres make up $2^{4+3}=2^7$ words, which is the total number of messages! The total space is saturated by non-overlapping Hamming spheres, and this is an example of a *perfect* code.

This construction looks like magic, but it was soon realized that a more transparent description existed in terms of linear algebra.

2 Canonical Description of Linear Codes

Classical codes are characterized by three parameters, n the length of the codewords (in bits), $k < n$ the length of the messages, and d the *minimum* distance (number of flips for binary) between codewords, assembled in the symbol $[\,n,\, k,\, d\,]$.

The just-discussed Hamming code is a $[\,7,\, 4,\, 3\,]$ code, described by the (4×7) *Generator Matrix*,

$$G = (I_4|A) = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix},$$

written here in *standard form*. The corresponding (3×7) *Parity Check Matrix*,

$$H = (-A^T|I_3) = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, \tag{2}$$

satisfies by construction

$$GH^T = I_4 A - A I_3 = 0, \quad (\mathrm{mod}\ 2). \tag{3}$$

This parity check matrix describes the $(7,4)$ code, since it can be obtained from the matrix of Eq(**??**) by permuting and/or adding rows and/or columns. What matters is the minimum number of linearly independent columns in H.

The generator inserts the $2^{k=4}$ bits messages into the $2^{n=7}$ codewords,

$$\mathbf{x} = [x_1, x_2, x_3, x_4] \;\longrightarrow\; w = \mathbf{x}\,G = [x_1, x_2, x_3, x_4, (x_1{+}x_2{+}x_3), (x_2{+}x_3{+}x_4), (x_1{+}x_2{+}x_4)].$$

The codewords w satisfy,

$$H\,w = 0 \pmod 2.$$

Since $(GH^T)^T = HG^T = 0$, the columns of G^T define four independent codewords,

$$w_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}, \quad w_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \quad w_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \quad w_4 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}.$$

The minimum distance (number of flips) between any of these is three, the code's distance. The rest of the codewords are linear combinations,

$$w = \sum_{i=1}^{4} \alpha_i w_i,$$

with α_i valued over Z_2. Including the zero codeword, there are $2^4 = 16$ codewords.

Flip Errors

Consider the transmission $w \implies w'$. The recipient checks for errors in transmission by computing the error syndrome Hw'. If there is no error, $Hw' = 0$, but if $Hw' \neq 0 \pmod 2$, errors have occurred.

The second step is to identify the locations of the flip(s) in the binary chain. Hamming chose H such that each of the seven locations of a *single* flip correspond to a unique error syndrome:

$$Hw' = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \quad \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \quad \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

with the flip occurring in the first, second,..., seventh positions respectively. However the error syndromes cannot unambiguously locate multiple flips.

Dual Codes

A code is said to be linear if a linear combination of codewords is itself a codeword. Any classical linear code C contains a dual code $C^\perp$, with the parity check and generator matrices interchanged. Transposition

$$HG^T = 0 \quad \rightarrow \quad GH^T = 0,$$

defines the dual code parity check and generator matrices,

$$C^\perp : \qquad H^\perp = G, \qquad G^\perp = H, \qquad H^\perp (G^\perp)^T = 0.$$

For $C = [7,4,3]$, the three codewords of $C^\perp$ are the columns of $(G^\perp)^T = H^T$,

$$w_1^\perp = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad w_2^\perp = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad w_3^\perp = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

The C and $C^\perp$ codewords are orthogonal to one another,

$$w_i \cdot w_a^\perp = 0, \quad i = 1,2,3,4, \quad a = 1,2,3 \pmod 2.$$

$C^\perp$ has $d^\perp = 4$. Note that $C^\perp \subseteq C = [7,4,3]$.

Note that the matrix A is not square. However an extended code can be generated by adding to A a fourth column. G becomes a (8×4) matrix and the extended code is $[8,4,4]$. It is self-dual.

Since C is linear (linear combinations of codewords are codewords), we can change the codeword basis. The new basis,

$$\begin{aligned} w_1' &\equiv w_1 + w_2 + w_3 = (1\,1\,1\,0\,1\,0\,0)^T, \quad w_2' \equiv w_2 + w_3 + w_4 = (0\,1\,1\,1\,0\,1\,0)^T, \\ w_3' &\equiv w_1 + w_2 + w_4 = (1\,1\,0\,1\,0\,0\,1)^T, \quad w_4' \equiv w_1 + w_3 + w_4 = (1\,0\,1\,1\,0\,0\,0)^T, \end{aligned}$$

yields a new generator matrix without affecting $G_{new} H^T = 0$,

$$G_{new} = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}. \tag{4}$$

It has been arranged so that its first three rows are the parity check matrix,

$$G_{new} = H_{new}^\perp = \begin{pmatrix} H \\ s^T \end{pmatrix},$$

and s is a codeword of weight three. Half the codewords in the $[7,4,3]$ Hamming code have even weight.

Any element of the dual code $u \in C^\perp$, satisfies $H_{new}^\perp u = 0$, so that

$$u \in C^\perp \quad \rightarrow \quad Hu = 0 \text{ and } s^T u = 0.$$

On the other hand for elements in C but not in $C^\perp$, $H_{new}^\perp u \neq 0$, but since $u \in C$, $Hu = 0$, so that necessarily $s^T u \neq 0$:

$$u \in C, \ \notin C^{\perp} \quad \rightarrow \quad Hu = 0 \quad \rightarrow \quad s^T u \neq 0.$$

C codewords split into two classes; those orthogonal to s which are in $C^{\perp}$, and those not orthogonal to s in $C/C^{\perp}$.

Pairs of C codewords, u, v are said to be *equivalent* if,

$$s^T u = s^T v \neq 0 \quad \longrightarrow \quad u \equiv v,$$

so that for each row of the generator matrix, we can assign two states under this equivalence. This will prove useful in constructing quantum codes.

For each codeword there is a Hamming spheres of "radius" h, the set of words that differ from the one codeword by no more than h flips. They will not overlap as long as d, the minimum distance between codewords is greater than $2h$. For $d = 3$, the best we can have is $h = 1$, leading to single error correction as in the $[7, 4, 3]$ code. If d is even, the maximum radius is the largest integer less than $d/2$: a double flip error correction code requires Hamming spheres of radius two, so d needs to be at least five. The dual code $[7, 3, 4]$ is single error-correcting.

Consider a code with n binary entries, k parity checks and minimum distance d. The number of points in a Hamming sphere of radius h is given by

$$M(n, h) = 1 + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{h},$$

where h is the largest integer in $[d/2 - 1]$.

There are 2^k such spheres, yielding a total number of $2^k M(n, h)$ points, which determines the minimum number of words,

$$2^k M(n, h) \leq 2^n.$$

When the equality holds, the code is said to be a *Perfect Code*. Although the Hamming code is perfect, its dual $C^{\perp} = [7, 3, 4]$, with four parity checks, and 2^3 codewords is not.

3 Galois Fields

A *field* is a set of elements including 0 and 1, with commutative addition and multiplication laws. Each element has an inverse except the zero element. Without the zero, it has the algebraic structure of two Abelian groups under addition and multiplication.

A Galois field $GF(p)$is a finite set of elements with q elements 0, 1, 2, $\cdots$, $q-1$ which satisfy the field axioms, where q is any power of a prime, $q = p^n$. Addition and multiplication are performed (mod p).

The simplest Galois field is the binary $GF(2)$ with elements 0, 1, where 1 is its own inverse, we have considered so far.

When $q = p$ prime, Galois fields $GF(p)$ have p elements $0, 1, \cdots p-1$ in one to one correspondence with the integers. For instance $GF(3)$ has three elements0, 1, and 2 with (mod 3) multiplication:$2 \cdot 2 = 4 = 1$ (mod 3).

When q is the power of a prime, their construction is more subtle and their elements no longer correspond to integers.

For example, to construct $GF(2^2)$, one starts from $GF(2)$ with two elements, 0 and 1. Let x be in $GF(4)/GF(2)$, yielding by addition two new elements x and $x+1$. There are four second order polynomials over $GF(2)$, x^2, x^2+1, x^2+x, x^2+x+1. The first three have roots in $GF(2)$: $0, 1, 1$ respectively. The fourth $\pi(x) = x^2+x+1$ does not, and is called irreducible: $\pi(x) = 0 \;\rightarrow\; x \notin GF(2)$

To show that they close under multiplication, we use Galois's genial idea and define it multiplication (mod π), which allows to express x^2 in terms of lower powers of x; for instance $x^2 = -x-1 = x+1 \bmod(\pi)$, resulting in the multiplication table

	1	x	$x+1$
1	1	x	$x+1$
x	x	$x+1$	1
$x+1$	$x+1$	1	x

All elements are cubic roots of unity: $x^3 = xx^2 = x(x+1) = x^2+x = x^2+x+1-1 = 1$, $(x+1)^3 = x^3+3x^2+3x+1 = 1+3(x+1)+3x+1 = 6x+5 = 1$, (mod 2, $\pi(x)$). Hence we can set $x = \omega = \exp(\frac{2\pi i}{3})$. Note that ω satisfies $\omega^2+\omega+1 = 0$, $\omega^3 = 1$.

A less trivial is $GF(3^2 = 9)$ with several irreducible polynomials. Start from the *integral* elements in $GF(3)$, $0, 1, 2$, to which we add polynomials obtained by successive addition, leading to nine elements (three integral plus four non-integral) elements

$$GF(9): \quad \{0,\, 1,\, 2,\, x,\, x+1,\, x+2,\, 2x,\, 2x+1,\, 2x+2\}\ .$$

The reducible quadratic polynomials are $(x-1)^2, (x-2)^2, (x-1)(x-2), x(x-1), x(x-2), x^2$ to be evaluated (mod 3). There are three irreducible polynomials

$$\pi_1(x) \;=\; x^2+1\ , \qquad \pi_2(x) \;=\; x^2+2x+2\ , \qquad \pi_3(x) \;=\; x^2+x+2\ ,$$

without roots in $GF(3)$. Thankfully, the same multiplication table is determined using any of these irreducible polynomials; it is simplest to use $\pi_1(x)$, so that $x^2 = -1 \sim 2 \pmod{\pi_1}$. Labeling the non-zero elements as $a = 1, 2 \ldots, 8$, the multiplication table is,

a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
a_2	a_1	a_6	a_8	a_7	a_3	a_5	a_4
a_3	a_6	a_2	a_5	a_8	a_6	a_4	a_7
a_4	a_8	a_5	a_6	a_1	a_7	a_2	a_3
a_5	a_7	a_8	a_1	a_3	a_4	a_6	a_2
a_6	a_3	a_1	a_7	a_4	a_2	a_8	a_5
a_7	a_5	a_4	a_2	a_6	a_8	a_3	a_1
a_8	a_4	a_7	a_3	a_2	a_5	a_1	a_6

All elements of $GF(9)$ are roots of the polynomial $x^9 - x$, which generalizes to any $q = p^n$.

For $x \in GF(q = p^m)$, define the trace

$$\mathrm{Tr}(x) = x + x^p + x^{p^2} + \cdots + x^{p^m - 1}$$

is an element of Z_p, with nice properties

$$\mathrm{Tr}(x+y) = \mathrm{Tr}(x) + \mathrm{Tr}(y);\ \ \mathrm{Tr}(x^p) = \mathrm{Tr}(x)$$

x has a basis $(\beta_1, \beta_2, \ldots, \beta_m)$, with each β_a assuming p values $(1, 2, \ldots, p-1)$.

Basis for $GF(3^2)$. From $\alpha_1,\ \alpha_2$, one can generate all eight non-zero elements of $GF(9)$: $\alpha_1,\ \alpha_2,\ 2\alpha_1,\ 2\alpha_2,\ \alpha_1+\alpha_2,\ 2\alpha_1+\alpha_2,\ \alpha_1+2\alpha_2,\ 2\alpha_1+2\alpha_2$.

Basis for $GF(2^3)$: $\alpha_1,\ \alpha_2,\ \alpha_3,\ \alpha_1+\alpha_2,\ \alpha_1+\alpha_3,\ \alpha_2+\alpha_3,\ \alpha_1+\alpha_2+\alpha_3$.

4 Golay Codes

Shannon had mentioned in his seminal 1948 paper Hamming's $(7,4)$ code, which was not published until 1950 (when Hamming and Holbrook, who designed the "circuit", were awarded a patent).

The Swiss physicist Marcel Golay who had read Shannon's article quickly generalized the $(7,4)$ code to any prime p, and invented several new codes in his seminal 1949 paper[**?**]. His notation was slightly different, since it relied on Shannon's paper.

Golay's message is made up of seven binary entries, four of which, Y_3, Y_5, Y_6, Y_7 label the codewords, and the other three are check digit entries X_1, X_2, X_4, computed so that all three combinations,

$$X_1 + Y_3 + Y_5 + Y_7, \quad X_2 + Y_3 + Y_6 + Y_7, \quad X_4 + Y_5 + Y_6 + Y_7, \tag{5}$$

vanish (mod 2). Not all vanish If the message is corrupted, and their values (error syndrome) determine which binary entry has flipped. His seven binary entries have the codewords at the beginning and the last three are the "redundants". The matrix

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

vanishes on the codewords; otherwise the codeword is in error. It is clearly the same as Hamming's code, and there is a controversy because Golay was the first to publish it.

Golay goes beyond binary entries, and now the messages are strings of n $p-$ary entries, each with values, $0, 1, 2 \ldots, p-1$ for prime p.

Consider strings of length n with p^n words, including p^k codewords. A single-error Hamming sphere includes $1 + n(p-1)$ words. The code is perfect if $p^k(1 + n(p-1)) = p^n$, or

$$n = \frac{p^{(n-k)} - 1}{p - 1}. \tag{6}$$

Golay proceeds to construct by induction new matrices associated with these $p-$ary perfect codes. His algorithm, starting from a $(k \times n)$ matrix for a perfect one error-correcting code satisfying Eq.(**??**), generates a $(k' \times n')$ matrix for the perfect code with $n' = np + 1$, $k' = k + 1$ and $r' = n(p-1) + r$ binary checks, such that,

- its first k rows are made up of p rows of the original matrix, adding a zero in the last column.

- its last row is made up of n $(p-1)$'s, n $(p-2)$'s,... up to n zeros, followed by a one.

• $p = 2$ (Hamming-Golay code)

$$k = 1 \; n = 1, \qquad A = (1); \qquad\qquad k = 2 \;\; n = 3, \qquad A = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

$$k = 3, \;\; n = 7, \qquad A = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix},$$

• $p = 3$

$$k = 2 \;\; n = 4, \qquad \begin{pmatrix} 1 & 1 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix}$$

$$k = 3 \;\; n = 13, \qquad A = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 2 & 1 & 0 & 1 & 2 & 1 & 0 & 1 & 2 & 1 & 0 & 1 & 0 \\ 2 & 2 & 2 & 2 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

and so-on.

Golay noticed the mathematical identities,

$$\sum_{i=0}^{2} \binom{90}{i} = 2^{12}, \qquad \sum_{i=0}^{3} \binom{23}{i} = 2^{11},$$

which suggest two perfect binary ($p = 2$) codes, $(90, 78)$ and $(23, 12)$. The second identity shows that a set of non-overlapping Hamming spheres of radius 3 (errors) saturate binary words of length 12. It implies the perfect 3-error correcting (23,12) code. Its eleven check digits satisfy,

$$X_i + \sum_{j=1}^{12} a_{ji} Y_j \equiv 0, \;\; (\text{mod } 2) \quad \rightarrow \quad X_i = \sum_{j=1}^{12} a_{ji} Y_j \;\; (\text{mod } 2),$$

where a_{ij} are the binary elements of the (11×12) matrix,

$$A^T = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

The codewords which are its eigenvectors with zero eigenvalue, have at least seven ones. One can define the corresponding (23×11) generator matrix,

$$G = [\, I_{12} \,|\, A \,],$$

as a linear map from the 2^{23}-dimensional vector space to a 2^{11}-dimensional subspace.

Another identity

$$\sum_{i=0}^{2} (p-1)^i \binom{11}{i} = 3^5,$$

leads to a ternary $(p = 3)$ perfect two-error correcting code $(11, 6)$. The ternary elements are $1, 2, 3$ added and multiplied (mod 3). Its (5×6) matrix with ternary entries is given by,

$$A^T = \begin{pmatrix} 1 & 1 & 1 & 2 & 2 & 0 \\ 1 & 1 & 2 & 1 & 0 & 2 \\ 1 & 2 & 1 & 0 & 1 & 2 \\ 1 & 2 & 0 & 1 & 2 & 1 \\ 1 & 0 & 2 & 2 & 1 & 1 \end{pmatrix}$$

This is some of the contents of Golay's 1949 paper! This was followed by a method to construct error-correcting codes based on p^m.

Hamming codes generalize $[7, 4, 3]$ to entries over $GF(q)$. These q-ary codes with parameters,

$$[\, n,\, k,\, d \,] = [\, \frac{q^r - 1}{q - 1},\, \frac{q^r - 1}{q - 1} - r,\, d \,],$$

describe an infinite family of single error-correcting perfect codes with q^{n-r} codewords.

Steiner Systems

Steiner systems $S(p, q, r)$ with $p \leq q \leq r$ such that each p-set occurs **only once** in all possible q-sets made out of r integers.

Projective geometries with $s^2 + s + 1$ points, and $s + 1$ lines, (s prime) are Steiner systems $S(2, s+1, s^2+s+1)$, because of the axiom of projective geometry which says that no two points can appear on more than one line. For $s = 2$ the smallest projective plane is the Fano plane.

Some Steiner systems are connected with Golay codes and Mathieu groups,

$$S(4, 5, 11) \quad \leftarrow \quad S(5, 6, 12), \qquad S(3, 6, 22) \quad \leftarrow \quad S(4, 7, 23) \quad \leftarrow \quad S(5, 8, 24).$$

5 BCH Codes

These are multiple error-correcting codes over $GF(q)$ invented independently by A. Hocquenghem (1959) and R. C. Bose with D. K. Ray-Chaudhuri (1960). Amusingly, early references assumed the Ray in Ray-Chaudhuri's name stood for Raymond, hence the error in the name- an assignation of error-correcting codes with an error! In 1960, I. S. Reed and G. Solomon invented a special case of BCH codes, with an easier decoding algorithm.

The ingredient behind these codes is the Vandermonde matrix,

$$V = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ a_1 & a_2 & \cdots & a_r \\ a_1^2 & a_2^2 & \cdots & a_r^2 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ a_1^{r-1} & a_2^{r-1} & \cdots & a_r^{r-1} \end{pmatrix},$$

whose determinant,

$$\det V = \prod_{i>j} (a_i - a_j)$$

does not vanish: the columns of the Vandermonde matrix are linearly independent.

The minimum distance between classical codewords is the number of non-zero entries between them as well as the minimum weight of the codewords. Consider a code with minimum weight d. Its codewords $\mathbf{x} = (x_1 x_2 \ldots x_n)^T$ satisfiy the parity check equation $H\mathbf{x} = 0$, that is

$$x_1 H_1 + x_2 H_2 + \cdots x_n H_n = 0$$

where H_i are the column vectors of $H = (H_1\, H_2\, \cdots\, H_n)$. Some of these column vectors are linearly dependent. Consider a codeword of weight $(d-1)$,

$$\mathbf{x} = (0 \cdots 0\, x_{i_1} 0 \cdots 0\, x_{i_2} 0\, \cdots 0\, x_{i_{d-1}} 0 \cdots 0)^T.$$

As a codeword it should satisfy,

$$x_{i_1} H_{i_1} + x_{i_2} H_{i_2} + \cdots + x_{i_{d-1}} H_{i_{d-1}} = 0\,?,$$

but the smallest weight is d, so this expression does not vanish; we conclude that there are $(d-1)$ linearly independent column vectors in H.

It behooves us to consider the $(d-1) \times n$ parity check matrix,

$$H = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & 2 & \dots & n \\ 1 & 2^2 & \cdots & n^2 \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ 1 & 2^{d-2} & \dots & n^{d-2} \end{pmatrix}$$

with entries over $GF(p)$, $n \geq d$ and $p \geq (n+1)$ prime.

Its virtue is that any $(d-1)$ columns of H form a $(d-1)\times(d-1)$ Vandermonde matrix, with non-zero determinant: any $(d-1)$ columns are linearly independent vectors, and it can be checked that any d columns are linearly dependent.

A parity matrix with $(d-1)$ linearly independent columns imply that d is the minimum distance between codewords. To avoid ambiguity Hamming spheres must not overlap. The centers of two Hamming spheres are d steps apart. Hence odd $d = 2t+1$ and even $d = 2t$ imply a t-error-correcting code:

H is the parity check matrix of the $[\, n,\, n-d+1,\, d\,]$ BCH code.

For example, the parity check matrix for a code over $GF(11)$

$$H = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & 2 & \dots & 10 \\ 1 & 2^2 & \cdots & 10^2 \\ 1 & 2^3 & \cdots & 10^3 \end{pmatrix}$$

contains at most four linearly independent vectors: its code has minimum distance $d = 5$ and corrects at most two errors.

The second example is a code also on $GF(11)$ with

$$H = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & 2 & \dots & 10 \\ 1 & 2^2 & \cdots & 10^2 \\ 1 & 2^3 & \cdots & 10^3 \\ 1 & 2^4 & \cdots & 10^4 \end{pmatrix}$$

With five linearly independent vectors, its distance is $d = 6$, and it corrects up to three errors.

Error Syndrome

In general the codewords $(x_1 x_2 \cdots x_n)$ must satisfy

$$\sum_{i=1}^{n} i^j x_i = 0 \pmod p, \quad j \;=\; 0, 1, 2 \cdots, d-2.$$

We assume that $d = 2t+1$ is odd, where t is the maximum number of correctible errors.

Now suppose that a message word has been transmitted $\mathbf{x} \implies \mathbf{y}$. To check for errors, the recipient computes the error syndrome,

$$\mathbf{S} = \mathbf{y} H^T,$$

where $\mathbf{S} = (S_1, S_2, \ldots, S_{2t})$. Assume a maximum number of errors at locations $X_1, X_2, \ldots, X_t$, with errors $\Delta_1, \Delta_2, \ldots, \Delta_t$, so that

$$S_j \;=\; \sum_{i=1}^{t} \Delta_i X_i^{j-1}, \qquad j = 1, 2, \ldots, 2t.$$

The mathematical challenge is to identify from $\mathbf{S}$ both the location and "sizes" of the errors.

A similar set of equations was solved by no less than Ramanujan in 1912 who considers the function,

$$\varphi(z) = \sum_{i=1}^{t} \frac{\Delta_i}{1 - zX_i},$$

where z is a real variable. By expanding each denominator,

$$\frac{1}{1 - zX_i} = (1 + zX_i + z^2 X_i^2 + \cdots),$$

one generates the the infinite series

$$\varphi(z) = \sum_{i=1}^{\infty} z^{i-1} S_i,$$

where the coefficients of the first $2t$ terms are the error syndromes.

On the other hand, the same function can be expressed by reducing it to a common denominator as the ratio of two polynomials of order $t-1$ and t in terms of unknown functions A_i and B_i, $i = 1, 2 \cdots t$,

$$\varphi(z) = \frac{\sum_{j=1}^{t} A_j z^{j-1}}{\sum_{k=0}^{t} B_k z^k}, \tag{7}$$

with $B_0 \equiv 1$. Comparison of these two expressions of $\varphi(z)$ yields,

$$\sum_{k=0}^{t} B_k z^k \sum_{i=1}^{\infty} z^{i-1} S_i = \sum_{l=1}^{t} A_l z^{l-1}.$$

The left-hand side is an infinite series in z, while the right-hand side terminates at z^{t-1}. Equating powers of z, yields two sets of equations. The first expresses A_j in terms of S_i and B_k,

$$A_j = \sum_{i=0}^{j-1} S_{j-i} B_i, \qquad j = 1, 2, \ldots, t. \tag{8}$$

The second set of equations do not contain the A_i, and yields the coefficients of the extra powers in the left-hand side to zero,

$$\sum_{i=0}^{t} S_{t+j-i} B_i = 0,$$

which hold for $j = 1, 2, \cdots, \infty$. The first t of these equations determine B_i in terms of the rest of the syndromes S_1, S_2, $\ldots$, S_{2t}. One should check these equations for linear independence by computing the determinant of the matrix $\mathbf{S}$. The rest are self consistency equations of no interest here.

Knowing the B_i determine the zeros of the polynomial in the denominator of Eq.(**??**), which are nothing but the inverses of the locations of the errors. We then compute the A_i from Eq.(**??**) in order to get the size of each error. The final step is to return to Eq.(**??**) and rewrite it in its original form, and we are done!

An important consideration is the ease of the decoding computation, and people have since developed more efficient decodings, but I could not help discussing Ramanujan's method.

More notation: the error and error-evaluating polynomials

$$\sigma(z) = \prod_{1}^{t} (1 - z X_i), \qquad \omega(z) = \sum_{1}^{t-1} A_i z^i,$$

are useful tools in organizing the decoding.

6 Cyclic Codes

Consider codewords of length n, $(f_0 f_1 \cdots f_{n-1})$. An elegant way to describe them is by polynomials of degree less than n,

$$f(x) = f_0 + f_1 x + \cdots f_{n-1} x^{n-1}. \tag{9}$$

Polynomials are fun to play with; they can be added and multiplied at will, and form a ring (field without multiplicative inverses), However the world of polynomials of degree less than n is naturally described by general polynomials with multiplication defined modulo some polynomial of degree n. We have encountered this type in Galois's construction of $GF(p^n)$.

Two polynomials $f(x)$ and $g(x)$ are congruent (mod $k(x)$) if

$$f(x) \approx g(x) \mod k(x) \quad \longrightarrow \quad f(x) = g(x)k(x).$$

If we choose $k(x) = x^n - 1$ all congruent polynomials have degree less than n, because x^n can be replaced by 1. For instance, (mod $x^n - 1$) multiplication by x

$$xf(x) = f_0 x + f_1 x^2 + \cdots + f_{n-2} x^{n-1} + f_{n-1} x^n = +f_{n-1} + f_0 x + f_1 x^2 + \cdots + f_{n-2} x^{n-1},$$

maps the codeword $(f_0 f_1 \cdots f_{n-1})$ to another codeword $(f_{n-1} f_0 \cdots f_{n-2})$, which is a cyclic permutation. This multiplication rule is at the heart of Cyclic Codes. Not all codes are cyclic but some familiar codes can be written in cyclic form, for example the $[7, 4, 3]$ code after rearrangement of the codeword coordinates.

Cyclic Code Factoids

Clearly any sum of two polynomials of degree less than n is a polynomial of the same ilk: the code is linear. Also for any such polynomial $f(x)$, its product with another polynomial g(x) is also a polynomial of order less than n.

- Let R_n contain all the polynomials of rank less than n, (mod 2). Take any polynomial of in R_n; its products by all the polynomials in R_n mod $(x^n - 1)$ generate (in general redundantly) all the codewords in the cyclic code.

- There is a unique polynomial with highest degree coefficient is one (monic) of smallest degree, which is a factor of $x^n - 1$.

Cyclic codes can be defined in terms of one polynomial of lesser degree r,

$$g(x) = g_0 + g_1 x + \cdots + g_r x^r,$$

which describes a codeword of length n, with $(n-r-1)$ zero entries, $(g_0 g_1 \cdots g_r 0 \cdots 0)$. The $(n - r - 1)$ codewords obtained by cyclic permutations $(0 g_0 g_1 \cdots g_r 0 \cdot 0)$, $(00 g_0 g_1 \cdots g_r 0 \cdot 00)$, etc..., are linearly independent. They are assembled in a $(n - r) \times n$ generator matrix

$$G = \begin{pmatrix} g_0 & g_1 & g_2 & \cdot & \cdot & \cdot & g_r & 0 & 0 & \cdot & 0 \\ 0 & g_0 & g_1 & g_2 & \cdot & \cdot & g_{r-1} & g_r & 0 & \cdot & 0 \\ 0 & 0 & g_0 & g_1 & \cdot & \cdot & \cdot & g_{r-1} & g_r & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & 0 & \cdot & g_0 & g_1 & \cdot & \cdot & \cdot & g_{r-1} & g_r \end{pmatrix}.$$

Its rows represent codewords of the form $g(x), xg(x), \ldots x^{n-r-1}g(x)$, and since they are linearly independent they serve as the generators of the code.

It is not too hard to see that multiplication of $g(x)$ by any other polynomial will generate a linear combination of these $(n-r)$ codewords.

Consider the polynomial $h(x)$ of degree $k = (n-r)$ which satisfies

$$x^n - 1 = g(x)h(x).$$

Since codeword are of the form $w(x) = p(x)g(x) \pmod{(x^n - 1)}$, necessarily

$$w(x)h(x) = p(x)g(x)h(x) = 0, \quad \bmod\ (x^n - 1).$$

It is easy to show that the reverse holds true, that is from $w(x)h(x) = 0$, $w(x) = p(x)g(x)$, follows. Naturally, $h(x)$ is called the check polynomial of the cyclic code. Let

$$h(x) = h_0 + h_1 x + h_2 x^2 + \cdots + h_k x^k.$$

Its reciprocal polynomial is defined as,

$$\bar{h}(x) = h_k + h_{k-1}x + h_{k-2}x^2 + \cdots + h_0 x^k = x^k h(x^{-1}).$$

The dual code is generated not by $h(x)$ but by $\bar{h}(x)$. To see this, consider the identity,

$$x^{23} - 1 = (x-1)g_1(x)g_2(x),$$

with

$$g_1(x) = x^{11}+x^{10}+x^6+x^5+x^4+x^2+1, \qquad g_2(x) = x^{11}+x^9+x^7+x^6+x^5+x+1, \pmod{2}.$$

Both $g_1(x)$ and $g_2(x)$ can serve as generators since each divides $x^{23} - 1$. Note that $g_2(x) = x^{11}g_1(x^{-1})$ is the reciprocal polynomial.

$g_1(x)$ generates the $[23, 12, 7]$ binary Golay code is cyclic, while $g_2(x) = \bar{g}_1(x)$ generates its dual. They are different, but the extended Golay code $[24, 12, 8]$ is self-dual.