Example of simple BCH error-correcting coding

Problem: Design a 2-error correcting BCH code C of length 13 over the field GF(27). Find a non-zero code word in C, impose two errors and correct using an efficient algorithm.

Definition: Let β be a primitive root of the polynomial $x^n - 1$. If we use the generator polynomial $g(x)$ such that β, β^2, ..., β^{d-1}, are all roots of $g(x)$, such a code is then called a BCH-code with design distance d.

Part 1: factorization into irreducible factors and listing roots of those factors.

In GF(3) holds

$$x^{13}-1 = (x+2)(x^3+2x+2)(x^3+x^2+2)(x^3+x^2+x+2)(x^3+2x^2+2x+2).$$

Denote these polynomials as follows

$$\begin{aligned} f_1(x) &:= x+2, \\ f_2(x) &:= x^3+2x+2, \\ f_3(x) &:= x^3+x^2+2, \\ f_4(x) &:= x^3+x^2+x+2, \\ f_5(x) &:= x^3+2x^2+2x+2. \end{aligned}$$

We will now work in the algebraic field extension of GF(3) such that $x^{13}-1$ is factored into linear polynomials $(x-\beta_i)$ where β_i are primitive roots of 1. Such a field is GF(27).

Let β be a zero of f_2, i.e. $f_2(\beta) = \beta^3 + 2\beta + 2 = 0$. It also follows that β is a root of $x^{13} - 1$ and thus

$$\beta^{13} = 1. \tag{1}$$

Notorically known Lemma says that in the field $\mathbb{F}$ of characteristics p is for each $a, b \in \mathbb{F}$ and $k \in \mathbb{N}$

$$(a+b)^{p^k} = a^{p^k} + b^{p^k}. \tag{2}$$

Because in GF(27) is $2^3 = 2$, we have

$$f_2(\beta^3) = (\beta^3)^3 + 2\beta^3 + 2 = (\beta^3 + 2\beta^3 + 2)^3 = 0$$

and we conclude that β^3 is a zero of f_2. Using the same argument β^9 is as well a zero of f_2.

Let us compute $f_4(\beta^2)$. We know that $\beta^3 = -2\beta - 2 = \beta + 1$.

$$\begin{aligned} f_4(\beta^2) &= \beta^6 + \beta^4 + \beta^2 + 2 \\ &= (\beta+1)^2 + \beta(\beta+1) + \beta^2 + 2 \\ &= \beta^2 + 2\beta + 1 + \beta^2 + \beta + \beta^2 + 2 = 0. \end{aligned}$$

We can see that β^2 is a zero of f_4. Using the identity (2) in the similar manner as presented, we find that β^6 is also a zero of f_4. Assume that β^5 is a zero of f_i, then using (1) and (2) is also $(\beta^5)^3 = \beta^{15} = \beta^2$ a zero of f_i. We know that β^2 is a zero of f_4 and thus also β^5 must be a zero of f_4.

Furthermore

$$\begin{aligned} f_3(\beta^4) &= \beta^{12} + \beta^8 + 2 \\ &= (\beta+1)^4 + \beta^2(\beta+1)^2 + 2 \\ &= \beta^4 + 4\beta^3 + 2\beta^2 + 4\beta + 1 + 4\beta^2 + \beta^4 + 2\beta^3 + \beta^2 + 2 = 0 \\ f_3(\beta^{12}) &= 0 \end{aligned}$$

Using (1) and (2) we find that β^7, $(\beta^7)^3 = \beta^8$, $(\beta^8)^3 = \beta^{11}$ are zeros of the same f_i and i must be thus 5. Also β^{10} and β^4 are zeros of the same polynomial.

We can conclude this part stating that f_2 has zeros β, β^3, β^9; f_3 has zeros β^4, β^{10}, β^{12}; f_4 has zeros β^2, β^5, β^6 and finally f_5 has zeros β^7, β^8 and β^{11}.

Part 2: Designing the code

The powers of β in GF(27) forms the following table:

1	β	β^2	β^3	β^4	β^5	β^6	β^7	β^8	β^9	β^{10}	β^{11}	β^{12}	β^{13}
1	0	0	1	0	1	1	1	2	2	0	1	2	1
0	1	0	1	1	1	2	2	0	1	2	1	0	0
0	0	1	0	1	1	1	2	2	0	1	2	1	0

We want to develop a code C with the design distance $d = 5$. According to the definition of BCH codes, we shall take as a generating polynomial $g(x)$ product of those f_i such that β^1,...,β^4 are zeros of this product. Put

$$g(x) = f_2(x) \cdot f_3(x) \cdot f_4(x) = 2 + x^3 + x^4 + 2x^6 + x^7 + 2x^8 + x^9.$$

Because C is a cyclic code over GF(3), the generator matrix G can be written immediately

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

This matrix can be transformed into a canonical form $G_c = (I\ A)$:

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

From this we get a canonical parity-check matrix $H_c = (-A^T\ I)$

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

Back to the BCH codes. According to the definition, the parity-check matrix H of the code C has a form

$$H = \begin{pmatrix} 1 & \beta & \beta^2 & \beta^3 & \beta^4 & \beta^5 & \beta^6 & \beta^7 & \beta^8 & \beta^9 & \beta^{10} & \beta^{11} & \beta^{12} \\ 1 & \beta^2 & \beta^4 & \beta^6 & \beta^8 & \beta^{10} & \beta^{12} & \beta & \beta^3 & \beta^5 & \beta^7 & \beta^9 & \beta^{11} \\ 1 & \beta^3 & \beta^6 & \beta^9 & \beta^{12} & \beta^2 & \beta^5 & \beta^8 & \beta^{11} & \beta & \beta^4 & \beta^7 & \beta^{10} \\ 1 & \beta^4 & \beta^8 & \beta^{12} & \beta^3 & \beta^7 & \beta^{11} & \beta^2 & \beta^6 & \beta^{10} & \beta & \beta^5 & \beta^9 \end{pmatrix}.$$

If we substitute vectors from GF(27), we get

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

It can be checked that a vector space generated by H has a dimension 9 and is obviously the same as the vector space generated by H_c.

We conclude that we have designed a cyclic linear code C of the length 13 and dimension 3, over GF(3) with the generator matrix G_c and the parity-check matrix H_c. It is also a BCH code over GF(27) of the design distance 5 with the parity-check matrix H.

Part 3: characteristics of the code.

Length: 13	Dimension: 4
Information symbols: 9	Check symbols: 4
Rate: 9/13	Designed distance: 5
#code words: $\lvert\mathbb{F}_3\rvert^4 = 81$	Capable of correcting: 2 errors

Hilbert bound says that

$$M_3(13, d) \leq \frac{|\mathbb{F}_3|^{13}}{|B_e(w)|} = \frac{3^{13}}{\binom{13}{0} + \binom{13}{1} + \binom{13}{2}} = \frac{3^{13}}{1 + 13 + 78} < 17329.$$

Because $M_3(13, k = 4, d) < 3^4 = 81$, $M_3(13, k = 5, d) < 3^5 = 243$, we see, that the dimension of the code could be greatly improved.

Part 4: theory behind decoding.

This section is written rather briefly, introducing the theory is not the purpose of this paper.

Let us assume that the design distance is d, and that e errors ($e \leq 1/2d$) have occurred when transmitting certain codeword. Let $\sigma(z)$, $\omega(z)$ be polynomials

$$\sigma(z) = \prod_i (1 - \beta^i z) \quad \text{and} \quad \omega(z) = \sum_i E_i \beta^i z \prod_{j \neq i} (1 - \beta^j z),$$

where i and j run through the positions of the errors and E_i is the value of error i. The formal derivative of $\sigma(z)$ is

$$\sigma'(z) = \sum_i -\beta^i \prod (1 - \beta^j) z.$$

If we let $z = \beta^{-i}$ and divide ω by σ', we get

$$E_i = \beta^i \frac{\omega(\beta^{-i})}{\sigma'(\beta^{-i})},$$

where we still consider only those i where errors have occurred, or at least where $\sigma(\beta^{-i}) = 0$ so in both σ' and ω all terms but one disappear.

Now, we get

$$\epsilon(z) = \frac{\omega(z)}{\sigma(z)} = \sum_i \frac{E_i \beta^i z}{1 - \beta^i z} = \sum_i E_i \sum_{j=1}^{\infty} \beta^{ij} z^j = \sum_{j=1}^{\infty} z^j \sum_i E_i \beta^{ij}.$$

Because $\sum_i E_i \beta^{ij}$ equals to the inner product of the received word with the jth row of the check matrix, exactly $d - 1$ coefficients of $\epsilon(z)$ can be found. The inner product must be equal to zero for a correct word. Because ω and σ are at most of degree e and have constant terms 0 and 1, respectively, we only have $2e \leq d - 1$ unknowns. We find those by solving the equation

$$\omega(z) = \sigma(z)\epsilon(z).$$

This gives a system of $d - 1$ linear equations in the unknown coefficients of σ and ω.

Part 5: find a non-zero code word in C, impose two errors and correct.

In this part, we index elements from zero. The first row of a matrix is thus in our notation 0th row.

For simplicity assume that a word represented by the generating polynomial $g(x) = g(x_0x_1 \dots x_{12})$ was sent. We shall call this vector s,

$$s = (2\ 0\ 0\ 1\ 1\ 0\ 2\ 1\ 2\ 1\ 0\ 0\ 0).$$

Furthermore assume that during the transition two errors occurred, on positions 3 and 4, i.e. bits x_3 and x_4 were changed. We have thus received for instance the word

$$s' = (2\ 0\ 0\ 2\ 0\ 0\ 2\ 1\ 2\ 1\ 0\ 0\ 0).$$

Based on the theory, we want to compute the coefficients of the polynomial

$$\epsilon(z) = \sum_{j=1}^{\infty} \left(z^j \cdot \sum_i E_i \beta^{ij} \right),$$

where E_i is the value of error on position i. The j-th coefficient is equal to the inner product of s' and j-th row h_j of the parity-check matrix H. With a little abusive notation, we denote this coefficient $\epsilon(\beta^{j+1})$. When calculating the coefficient we can cheat and use the fact that $s \cdot h_j = 0$ for each $j = 0, \dots, 3$. We have to consider only the third column where 1 was during transition changed to 2 and the fourth column, where one was changed to 0. It means that $s' \cdot h_j = h_{j,3} - h_{j,4} = h_{j,3} + 2h_{j,4}$. A computer cannot cheat since it does not know where the position of errors are, but the results that it gets are naturally the same.

$$\begin{aligned}
\epsilon(\beta) &= \beta^3 - \beta^4 = \beta^3 + 2\beta^4 = 2\beta^{12} \\
\epsilon(\beta^2) &= \beta^6 - \beta^8 = \beta^6 + 2\beta^8 = 2\beta^5 \\
\epsilon(\beta^3) &= \beta^9 - \beta^{12} = \beta^9 + 2\beta^{12} = 2\beta^{10} \\
\epsilon(\beta^4) &= \beta^{12} - \beta^3 = \beta^{12} + 2\beta^3 = \beta^6
\end{aligned}$$

Using the key equation $\omega(z) = \sigma(z) \cdot \epsilon(z)$, we will find the error positions and will be able to correct the transition. Expanding the equation we get

$$\omega_1 z + \omega_2 z^2 = (1 + \sigma_1 z + \sigma_2 z^2)(2\beta^{12} z + 2\beta^5 z^2 + 2\beta^{10} z^3 + \beta^6 z^4).$$

We shall decompose this into the following equations

$$\begin{aligned} \omega_1 &= 2\beta^{12} \\ \omega_2 &= 2\beta^5 + 2\sigma_1\beta^{12} \\ 0 &= 2\beta^{10} + 2\sigma_1\beta^5 + 2\sigma_2\beta^{12} \\ 0 &= \beta^6 + 2\sigma_1\beta^{10} + 2\sigma_2\beta^5. \end{aligned}$$

Solving the linear system in four variables, we get that $\omega_1 = 2\beta^{12}$, $\omega_2 = 0$, $\sigma_1 = 2\beta^6$, $\sigma_2 = \beta^7$.

Based on the theory, we write:

$$\begin{aligned} \sigma(z) &= 1 + 2\beta^6 z + \beta^7 z^2 = (1 - \beta^3 z)(1 - \beta^4 z) \\ \sigma'(z) &= 2\beta^6 + 2\beta^7 z \\ \omega(z) &= 2\beta^{12} z^2 \end{aligned}$$

The roots of σ are β^{-3} and β^{-4} and we conclude that errors occurred on positions 3 and 4, i.e. s'_3 and s'_4 have to be corrected. To calculate the syndrome E_i, we will use the formula

$$\begin{aligned} E_i &= -\beta^i \frac{\omega(\beta^{-i})}{\sigma'(\beta^{-i})} \\ E_3 &= -\beta^3 \frac{\omega(\beta^{-3})}{\sigma'(\beta^{-3})} = -\beta^3 \frac{2\beta^6}{2\beta^6 + 2\beta^4} = -\beta^3 \frac{\beta^6}{2\beta^{12}} = \beta^{-3} \\ E_4 &= -\beta^4 \frac{\omega(\beta^{-4})}{\sigma'(\beta^{-4})} = -\beta^4 \frac{2\beta^4}{2\beta^6 + 2\beta^3} = -\beta^4 \frac{\beta^4}{\beta^{12}} = 2\beta^{-4} \end{aligned}$$

The syndrome is thus

$$s'' = (0\ 0\ 0\ 1\ 2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0),$$

and if we subtract it from the received word

$$s' = (2\ 0\ 0\ 2\ 0\ 0\ 2\ 1\ 2\ 1\ 0\ 0\ 0),$$

we get

$$s = (2\ 0\ 0\ 1\ 1\ 0\ 2\ 1\ 2\ 1\ 0\ 0\ 0),$$

exactly what was sent. This concludes the exercise.