Modern Computer Algebra

Joachim von zur Gathen, Jürgen Gerhard

Book DOI: http://dx.doi.org/10.1017/CBO9781139856065

Online ISBN: 9781139856065

Hardback ISBN: 9781107039032

Chapter

7 - Application: Decoding BCH codes pp. 209-216

Chapter DOI: http://dx.doi.org/10.1017/CBO9781139856065.009

Cambridge University Press

7
Application: Decoding BCH codes

Coding theory deals with the detection and correction of transmission errors. The scenario is that a message m is sent over a transmission channel, and due to noise on the channel some of the symbols in the received message r are different from those in m. How can we correct them?

A simple strategy is to send m three or five times and take a majority vote on each symbol. If errors occur too frequently, then this may not help much, but the usual assumption is that errors occur only with fairly small probability, and then this strategy will give an erroneous result only with much smaller probability than accepting r as is.

However, the cost (= length) of transmission has increased by a factor of three or five. The fundamental task of coding theory is to see whether small error probability can be achieved at reasonable cost. The basic framework of this theory was established in the pioneering work of Shannon (1948). Error correcting codes are employed in numerous situations, from computer networks to satellite TV, digital telephony, and the technology that make CDs so remarkably resistant against scratches. They must not be confused with *cryptography*, the art of sending secret messages that only the intended receiver can read (see Section 20).

It turns out that the tools of algebra provide many useful codes. We describe a particular class of such codes. Let $\mathbb{F}_q$ be a finite field with q elements, $n,k \in \mathbb{N}$, and $C \subseteq \mathbb{F}_q^n$ a k-dimensional linear subspace. C is called a **linear code** over $\mathbb{F}_q$. Any basis of C provides an isomorphism $\mathbb{F}_q^k \longrightarrow C$, and $\varepsilon\colon \mathbb{F}_q^k \longrightarrow C \subseteq \mathbb{F}_q^n$ is the **encoding map**. The number n is the **length** of C, k is its **dimension**, and the ratio $k/n \leq 1$ is the **rate** of C.

To transmit a message m, we first identify it with an element of $\mathbb{F}_q^k$. If, say, $q = 2$ and $k = 64$, and we want to transmit messages in ASCII, then each ASCII letter can be identified with an 8-bit string, and a block of 8 letters with a “word”

http://dx.doi.org/10.1017/CBO9781139856065.009

in $\mathbb{F}_2^{64}$. Now the simple code which sends each "word" three times has length 192, dimension 64, and rate $1/3$.

For an element $a = (a_1, \ldots, a_n) \in \mathbb{F}_q^n$, we denote by

$$w(a) = \#\{i : 1 \leq i \leq n, a_i \neq 0\}$$

its **Hamming weight**, and by

$$d(C) = \min\{w(a) : a \in C \setminus \{0\}\}$$

the **minimal distance** of C. Since C is a linear subspace, $w(a-b) \geq d(C)$ for all distinct $a, b \in C$. Our triple repetition code is $C = \{(a,a,a) \in \mathbb{F}_2^{192} : a \in \mathbb{F}_2^{64}\}$ and has minimal distance $d(C) = 3$.

On receiving a word $r \in \mathbb{F}_q^n$, it is decoded as $c \in C$ with $w(r-c)$ minimal. Since fewer errors are more probable, this is called **maximum likelihood decoding**. If less than $d(C)/2$ errors occurred in transmitting the word, then this will work correctly. If a single letter in $\mathbb{F}_q$ is received incorrectly with probability $\varepsilon \ll 1$, and errors occur independently, then this decoding procedure makes a mistake with probability no more than

$$\sum_{d(C)/2 \leq j \leq n} \binom{n}{j} \varepsilon^j (1-\varepsilon)^{n-j}.$$

One of the goals in coding theory is to make this probability small without decreasing the rate too much.

For example, $\varepsilon \approx 10^{-4}$ seems to be a reasonable value for transmissions over copper wires. In Table 7.1 below, we have a code C over $\mathbb{F}_2$ with dimension 8, length 15, and minimal distance 5. Then this error probability becomes $\approx 5 \cdot 10^{-8}$: a tremendous gain, at the cost of not even halving the transmission rate. This is much better than the triple repetition code mentioned above, which has error probability of about 10^{-5} and transmission rate $1/3$.

We now describe a popular class of codes, the BCH codes, together with an efficient way of implementing the decoding procedure.

Let $\mathbb{F}_q$ be a finite field and $f \in \mathbb{F}_q[x]$ be irreducible and monic with $\deg f = m$. Then $\mathbb{F}_{q^m} = \mathbb{F}_q[x]/\langle f \rangle$, and $\alpha = x \bmod f \in \mathbb{F}_{q^m}$ is a root of f (Lemma 4.5). Since $f(x^q) = f(x)^q$ for each $f \in \mathbb{F}_q[x]$, the elements $\alpha^q, \alpha^{q^2}, \ldots, \alpha^{q^{m-1}} \in \mathbb{F}_{q^m}$ are also roots of f. Furthermore, the α^{q^i} for $0 \leq i < m$ are all distinct (we will prove this in Section 14.10 when α is a primitive root of unity). Hence they are all roots of f, and $f = (x-\alpha)(x-\alpha^q)\cdots(x-\alpha^{q^{m-1}})$. The minimal polynomial of an element $\beta \in \mathbb{F}_{q^m}$ is the monic (nonzero) polynomial $f \in \mathbb{F}_q[x]$ of least degree such that $f(\beta) = 0$. It exists and is unique, and for all $g \in \mathbb{F}_q[x]$, we have $g(\beta) = 0$ if and only if $f \mid g$. These basic facts about finite fields are explained in Section 25.4.

EXAMPLE 7.1. (i) If $m = 1$, then the minimal polynomial of β is $f = x - \beta$.

(ii) The minimal polynomial of $\beta = x \bmod f \in \mathbb{F}_{q^m} = \mathbb{F}_q[x]/\langle f \rangle$ over $\mathbb{F}_q$ is f.

(iii) The polynomial $f = x^4 + x + 1 \in \mathbb{F}_2[x]$ is irreducible, and $\mathbb{F}_{16} = \mathbb{F}_2[x]/\langle f \rangle$ is a field with 16 elements. By (ii), the minimal polynomial of $\beta = x \bmod f \in \mathbb{F}_{16}$ is $x^4 + x + 1$. $\diamond$

DEFINITION 7.2. *An element $\beta \in \mathbb{F}_{q^m}$ is called a* ***primitive nth root of unity*** *if* $\gcd(n, q) = 1$ *and*

(i) $\beta^n = 1$, *and*

(ii) $\beta^k \neq 1$ *for* $0 < k < n$.

Thus a primitive nth root of unity is just an element of order n (Section 25.1) in the multiplicative group $\mathbb{F}_{q^m}^{\times} = \mathbb{F}_{q^m} \setminus \{0\}$. Such roots of unity will play a major role for the Fast Fourier Transform in Section 8.2. We are now in a position to say what a BCH code is.

DEFINITION 7.3. *Let $q = p^r$ for some prime p and let $n, \delta \geq 1$, β a primitive nth root of unity in some extension $\mathbb{F}_{q^m}$ of $\mathbb{F}_q$, and $g \in \mathbb{F}_q[x]$ the monic* lcm *of the minimal polynomials of $\beta, \beta^2, \ldots, \beta^{\delta-1}$. Then the vector space*

$$C = \sum_{0 \leq i < n - \deg g} x^i \overline{g} \cdot \mathbb{F}_q \subseteq \mathbb{F}_q[x]/\langle x^n - 1 \rangle = R \cong \mathbb{F}_q^n$$

is a ***BCH code****, denoted by* $\mathrm{BCH}(q, n, \delta)$. *Here $\overline{g} = (g \bmod x^n - 1) \in R$, and C is the ideal generated by $\overline{g}$ in R. The code C has length n and dimension $n - \deg g$, and g is its* ***generator polynomial****.*

The notation $\mathrm{BCH}(q, n, \delta)$ does not reflect the fact that the code depends on the choice of the primitive nth root of unity β, but the properties of the code (in particular, its minimal distance) are essentially independent of β. We will discuss in Section 14.10 how to construct BCH codes in general, and only give an example here.

EXAMPLE 7.4. We will construct all BCH codes of length 15 over $\mathbb{F}_2$. The factorization of $x^{15} - 1$ over $\mathbb{F}_2$ into irreducible factors is

$$x^{15} + 1 = (\underbrace{x + 1}_{f_1})(\underbrace{x^2 + x + 1}_{f_2})(\underbrace{x^4 + x^3 + x^2 + x + 1}_{f_3})(\underbrace{x^4 + x^3 + 1}_{f_4})(\underbrace{x^4 + x + 1}_{f_5}).$$

From the factorization of the **cyclotomic polynomials** Φ_k in $\mathbb{Z}[x]$ (Section 14.10) we find that $x^{15} - 1 = \Phi_1 \Phi_3 \Phi_5 \Phi_{15}$, where $\Phi_1 \equiv f_1$, $\Phi_3 \equiv f_2$, $\Phi_5 \equiv f_3$, and $\Phi_{15} \equiv f_4 f_5$

mod 2. We take $\mathbb{F}_{16} = \mathbb{F}_2[x]/\langle f_5 \rangle$, as in Example 7.1 (iii). For $\beta = x \bmod f_5 \in \mathbb{F}_{16}$, the elements $\beta^3, \beta^2, \beta, 1$ form a basis of $\mathbb{F}_{16}$ over $\mathbb{F}_2$, and

$$\mathbb{F}_{16} = \{a_3\beta^3 + a_2\beta^2 + a_1\beta + a_0 : a_3, a_2, a_1, a_0 \in \mathbb{F}_2\}.$$

We see that $\beta^3 \neq 1, \beta^5 = \beta^2 + \beta \neq 1$, and $\beta^{15} = 1$. This means that β is a primitive 15th root of unity. We only have to check the divisors of 15, because the order of β is a divisor of the order 15 of the multiplicative group $\mathbb{F}_{16}^\times$ of $\mathbb{F}_{16}$, by Lagrange's theorem.

Table 7.1 gives all BCH codes of length 15 over $\mathbb{F}_2$. ◇

δ	generator polynomial g	exponents i with $g(\beta^i) = 0$	$\dim C$	$d(C)$
1	1	$\varnothing$	15	1
$2,3$	f_5	$1,2,4,8$	11	3
$4,5$	$f_3 f_5$	$1,2,3,4,6,8,9,12$	8	5
$6,7$	$f_2 f_3 f_5$	$1,2,3,4,5,6,8,9,10,12$	5	7
$8,\ldots,15$	$f_2 f_3 f_4 f_5$	$1,\ldots,14$	1	15

TABLE 7.1: The BCH codes of length 15 over $\mathbb{F}_2$.

The parameter δ is called the **designed distance** of the BCH code. The next theorem shows that the minimal distance is at least as great.

THEOREM 7.5.
The minimal distance $d(C)$ of the code $C = \mathrm{BCH}(q, n, \delta)$ is at least δ.

PROOF. We identify $\mathbb{F}_q^n$ with $R = \mathbb{F}_q[x]/\langle x^n - 1 \rangle$ via

$$(a_{n-1}, \ldots, a_0) \longleftrightarrow a_{n-1}x^{n-1} + \cdots + a_1 x + a_0 \bmod x^n - 1.$$

Furthermore we have a primitive nth root of unity $\beta \in \mathbb{F}_{q^m}$ for some $m \geq 1$, and for $a \in R$ we have

$$\begin{aligned} a \in C &\iff a(\beta^i) = 0 \text{ for } 1 \leq i < \delta \\ &\iff \begin{pmatrix} \beta^{n-1} & \cdots & \beta^2 & \beta & 1 \\ \beta^{2(n-1)} & \cdots & \beta^4 & \beta^2 & 1 \\ \vdots & & \vdots & \vdots & \vdots \\ \beta^{(\delta-1)(n-1)} & \cdots & \beta^{2(\delta-1)} & \beta^{\delta-1} & 1 \end{pmatrix} \begin{pmatrix} a_{n-1} \\ \vdots \\ a_1 \\ a_0 \end{pmatrix} = 0. \end{aligned}$$

We denote the $(\delta - 1) \times n$-matrix above by B, and show that each $(\delta - 1) \times (\delta - 1)$ submatrix of B is nonsingular. From this the claim follows, because then for each $a \in C$ with $a \neq 0$ and $w(a) \leq \delta - 1$ we have $Ba \neq 0$.

http://dx.doi.org/10.1017/CBO9781139856065.009

For $0 \le i < n$, the $(n-i-1)$st column of B is

$$\begin{pmatrix} \beta^i \\ \beta^{2i} \\ \vdots \\ \beta^{(\delta-1)i} \end{pmatrix}.$$

If we divide it by β^i, we obtain

$$\begin{pmatrix} 1 \\ \beta^{i} \\ \vdots \\ \beta^{(\delta-2)i} \end{pmatrix},$$

which is a column of a Vandermonde matrix (Section 5.2). Hence each $(\delta-1) \times (\delta-1)$ submatrix of B is a Vandermonde matrix, where the columns are multiplied by some power of β. Since the β^i are pairwise distinct for $0 \le i < n$ and any power of β is nonzero, all such submatrices are nonsingular. □

Table 7.1 shows that the minimal distance of a BCH code can be strictly larger than the designed distance.

Now we will see how the decoding of a BCH code works. Let $C = \mathrm{BCH}(q,n,\delta)$ be given via β, and let δ be odd. Suppose that $c \in C$ is the transmitted and r the received word. We want to correct up to $t = (\delta-1)/2$ errors. Let

$$e = r - c = e_{n-1}x^{n-1} + \cdots + e_1 x + e_0 \bmod x^n - 1 \longleftrightarrow (e_{n-1}, \ldots, e_1, e_0)$$

be the error vector. Our assumption is that $w(e) \le t$. We define:

$$\begin{aligned} M &= \{i : e_i \neq 0\}, \text{ the positions where an error occurs,} \\ u &= \prod_{i \in M} (1 - \beta^i y) \in \mathbb{F}_q[y], \text{ the } \textbf{error locator polynomial}\text{, and} \\ v &= \sum_{i \in M} e_i \beta^i y \prod_{j \in M \setminus \{i\}} (1 - \beta^j y) \in \mathbb{F}_q[y]. \end{aligned}$$

Then $\#M \le t$, $\deg u \le t$, and $\deg v \le t$. If we know u and v, then the errors can be corrected in the following way. By evaluating u at $1, \beta^{-1}, \beta^{-2}, \ldots, \beta^{-n+1}$, we obtain M. If $i \in M$, then we use the following observations to calculate e_i (this is only necessary, of course, if $q > 2$). The formal derivative u' of u with respect to y (Section 9.3) is

$$u' = \sum_{i \in M} (-\beta^i) \prod_{j \in M \setminus \{i\}} (1 - \beta^j y).$$

Thus

$$v(\beta^{-i}) = e_i \prod_{j\in M\setminus\{i\}} (1-\beta^{j-i}) = -e_i\beta^{-i}u'(\beta^{-i}),$$

and hence

$$e_i = \frac{-v(\beta^{-i})\beta^i}{u'(\beta^{-i})}.$$

To compute u and v, we define

$$w = \frac{v}{u} = \sum_{i\in M} \frac{e_i\beta^i y}{1-\beta^i y} = \sum_{i\in M}\sum_{k\geq 1} e_i(\beta^i y)^k = \sum_{k\geq 1} y^k \sum_{i\in M} e_i\beta^{ki} = \sum_{k\geq 1} y^k e(\beta^k).$$

Since $c(\beta^k) = 0$ for $1 \leq k \leq \delta - 1$, we have $e(\beta^k) = r(\beta^k)$ for $1 \leq k \leq \delta - 1$. We can compute these values, because r is the received word, and hence compute w rem y^δ. So we have to solve the following problem: Given w rem y^δ, compute u and v.

It is possible to formulate this problem as a system of linear equations. On the other hand, v/u is just a $(t+1,t)$-Padé approximant to w. It is unique and can be computed with the Extended Euclidean Algorithm, as described in Section 5.9. The computation can be done with $O(\delta^2)$ operations in $\mathbb{F}_{q^m}$, and with $O^\sim(\delta)$ operations using the fast algorithms from Part II.

EXAMPLE 7.4 (continued). Let $g = f_5 = x^4 + x + 1 \in \mathbb{F}_2[x]$ from Example 7.4 be the generator polynomial of the code $C = \mathrm{BCH}(2,15,3)$ when we take $\beta = (x \bmod x^4 + x + 1) \in \mathbb{F}_{16} = \mathbb{F}_2[x]/\langle x^4 + x + 1\rangle$. Table 7.1 shows that $d(C) = 3$, and we can correct one error. Suppose that we have received

$$r = x^5 + x^4 + 1 \bmod x^{15} - 1 \in \mathbb{F}_2[x]/\langle x^{15} - 1\rangle.$$

Using $\beta^4 = \beta + 1$, we have

$$r(\beta) = \beta^5 + \beta^4 + 1 = \beta^2, \quad r(\beta^2) = \beta^{10} + \beta^8 + 1 = \beta + 1,$$

and hence

$$w = \sum_{k\geq 1} e(\beta^k)y^k = \sum_{k\geq 1} r(\beta^k)y^k \equiv (\beta+1)y^2 + \beta^2 y \bmod y^3. \tag{1}$$

Exercise 7.2 shows that the $(2,1)$-Padé approximant of w is

$$\frac{v}{u} = \frac{(\beta^3+\beta^2+\beta)y}{(\beta^3+\beta^2+\beta)y+\beta^3+\beta} = \frac{\beta^2 y}{\beta^2 y + 1},$$

so that $v = \beta^2 y$ and $u = \beta^2 y + 1$ in $\mathbb{F}_{16}[y]$. The only zero of u is at $y = \beta^{-2}$, and if we assume that at most one error has occurred, then this happened at position $i = 2$, and the original codeword was

$$c = x^5 + x^4 + x^2 + 1 \bmod x^{15} - 1.$$

In fact, we have $c = (x+1)g \bmod x^{15} - 1 \in C$. Expressing everything in terms of bit strings, the original message is $m = 00000000011$ of length eleven, encoded as $c^* = 000000000110101$ of length 15, using the isomorphism $\mathbb{F}_2^{11} \longrightarrow C$ given by the basis $x^{10}g, x^9g, \ldots, g$ of C. The received word is 000000000110001, and the decoding procedure discovers and corrects the error in the third last position, giving the receiver the correct word c^*, which then has to be converted back to the original message m. $\Diamond$

Notes. Coding theory was founded by Shannon (1948). There are many good texts available, among them Berlekamp (1984), MacWilliams & Sloane (1977), and van Lint (1982). The coding technology for CDs is described in detail in Hoffman, Leonard, Lindner, Phelps, Rodger & Wall (1991).

For arbitrary codes, it is not clear how to decode them efficiently, and, in fact, a sufficiently general version of the decoding problem is $\mathcal{NP}$-complete (Berlekamp, McEliece & van Tilborg 1978). BCH codes were discovered by Bose & Ray-Chaudhuri (1960) and independently by Hocquenghem (1959). Berlekamp (1984), already in the 1968 edition, and Massey (1965) discovered the decoding procedure for BCH codes, in a different formalism, and Dornstetter (1987) pointed out the relation to the Euclidean Algorithm.

Rabin (1989), Albanese, Blömer, Edmonds, Luby & Sudan (1994), and Alon, Edmonds & Luby (1995) describe *erasure codes*, a related class of codes which is used for communication over faulty networks that occasionally lose (or delay) packets (but do not change them).

Exercises.

7.1 Let F be a field, $k < n$ positive integers, and $u_1, \ldots, u_n \in F$ distinct. For $f \in F[x]$, let $\chi(f) = (f(u_1), \ldots, f(u_n)) \in F^n$, that is, χ is the evaluation map at $u_1, \ldots, u_n$. We define the linear code $C \subseteq F^n$ by $C = \{\chi(f) : f \in F[x], \deg f \leq k\}$. Show that C has minimal distance $n - k$.

7.2 Compute the $(2,1)$-Padé approximant to w from (1).

7.3 Determine generator polynomials and minimal distances of all BCH codes for $q = 2$ and $n = 7$. Hint: The polynomial $x^7 - 1 \in \mathbb{F}_2[x]$ factors into three irreducible polynomials

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

and $\beta = x \bmod x^3 + x + 1 \in \mathbb{F}_8 = \mathbb{F}_2[x]/\langle x^3+x+1\rangle$ is a primitive 7th root of unity.

7.4 Let $C = \mathrm{BCH}(2,7,3)$ be generated by $g = x^3 + x + 1 \in \mathbb{F}_2[x]$, and $\beta = x \bmod g$ be as in Exercise 7.3. Assuming that at most one error has occurred, decode the received words

$$r_1 = x^6 + x^5 + x^3 + 1 \bmod x^7 - 1, \quad r_2 = x^6 + x + 1 \bmod x^7 - 1.$$

Find a codeword $c \in C$ such that $d(r_2 - c) = 2$.

7.5$^{\longrightarrow}$ Let $q = 11$ and $n = 10$.

(i) Prove that $\beta = 2 \in \mathbb{F}_q$ is a primitive nth root of unity.

(ii) Show that the polynomial $x^{10} - 1$ splits into linear factors over $\mathbb{F}_q$.

(iii) Tabulate generator polynomials and minimal distances of all BCH codes for the above values of q, n, and β.

(iv) Let $C = \mathrm{BCH}(11,10,5)$. Check that the generator polynomial for C is $g = x^4 + 3x^3 + 5x^2 + 8x + 1$. Assuming that at most two errors have occurred, decode the received word

$$r = x^6 + 7x + 4 \bmod x^{10} - 1 \in \mathbb{F}_{11}[x]/\langle x^{10} - 1\rangle.$$