The BCH Bound

- **Theorem**: Let $C$ be cyclic of length $n$ with generator polynomial $g(x)$ over $\text{GF}(q)$. Let $m$ be the smallest integer such that $n | q^m - 1$ and let $\alpha \in \text{GF}(q^m)$ be a primitive $n$th root of unity. Then, if for some integers $b \geq 0$ and $\delta \geq 2$ all the elements

$$\alpha^b, \alpha^{b+1}, \ldots, \alpha^{b+\delta-2}$$

in $\text{GF}(q^m)$ are zeros of the code, it holds that $d_{\min} \geq \delta$.

$$\delta - 1 \text{ consecutive zeros } \Rightarrow d_{\min} \geq \delta$$
BCH Codes

- **Definition:** Consider a cyclic code $C$ of length $n$ over $\text{GF}(q)$, let $m$ be the smallest integer such that $n|q^m - 1$ and let $\alpha \in \text{GF}(q^m)$ be a primitive $n$th root of unity. Then $C$ is a *BCH code of designed distance* $\delta$ if for some $b \geq 0$ it has generator polynomial

$$g(x) = \text{lcm}\{p^{(b)}(x)p^{(b+1)}(x)p^{(b+\delta-2)}(x)\}$$

- A BCH code is said to be
  - *narrow sense* if $b = 1$
  - *primitive* if $n = q^m - 1$ ($\implies \alpha$ primitive in $\text{GF}(q^m)$)

- **Theorem:** A BCH code over $\text{GF}(q)$ of length $n$ and designed distance $\delta$ has $d_{\text{min}} \geq \delta$ and dimension $k \geq n - m(\delta - 1)$.

- In the special case $q = 2$, $b = 1$ and $\delta = 2\tau + 1$, it holds that $r = n - k \leq m\tau$

(since the $p^{(i)}(x)$'s have degree $\leq m$, and $p^{(2i)}(x) = p^{(i)}(x)$)

- **True minimum distance** $d_{\text{min}}$:
  - For $q = 2$, $b = 1$, $n = 2^m - 1$ and $\delta = 2\tau + 1$ the code has $d_{\text{min}} = 2\tau + 1$ if

$$\sum_{i=0}^{t+1} \binom{n}{i} > 2^{mt}$$

  - If $b = 1$ and $n = \delta p$ for some $p$, then $d_{\text{min}} = \delta$
  - If $b = 1$, $n = q^m - 1$ and $\delta = q^p - 1$ for some $p$ then, $d_{\text{min}} = \delta$
  - If $n = q^m - 1$ then $d_{\text{min}} \leq q\delta - 1$
Parity Check Matrix

- Assume narrow sense and primitive over GF(2) and \( \delta = 2 \tau + 1 \)
- Since \( g(\alpha^i) = 0 \) for \( i = 1, \ldots, \delta - 1 \), a valid parity check matrix is

\[
H_{BCH} = \begin{bmatrix}
1 & \alpha & \alpha^2 & \cdots & \alpha^{n-1} \\
1 & \alpha^3 & (\alpha^3)^2 & \cdots & (\alpha^3)^{n-1} \\
1 & \alpha^5 & (\alpha^5)^2 & \cdots & (\alpha^5)^{n-1} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & \alpha^{\delta-2} & (\alpha^{\delta-2})^2 & \cdots & (\alpha^{\delta-2})^{n-1}
\end{bmatrix}
\]

- That is, the second column = lowest-degree \( \alpha^i \)'s that correspond to different minimal polynomials
- To get the binary version: replace the \( \alpha^i \)'s with the column vectors from GF\(^m\)(2) that represent the coefficients of the polynomial \( \alpha^i \in GF(2^m) \)

- Gives \( m\tau \) binary rows, if \( m\tau > r \) reduce to get linearly independent rows

Examples

- **Binary Hamming code**: Narrow sense and primitive binary BCH code with \( n = 2^m - 1 \), for some \( m \geq 1 \), and \( g(x) = \) a primitive polynomial in GF(\( 2^m \)). Designed distance \( \delta = 3 = \text{true} \ d_{\text{min}} \)
- **Hamming code over GF(\( q \))**: A narrow sense and primitive BCH code, with \( m \) smallest integer such that \( n | q^m - 1 \), \( m \) and \( q - 1 \) relatively prime, and \( g(x) = \) primitive polynomial in GF(\( q^m \)). Designed distance \( \delta = 3 = \text{true} \ d_{\text{min}} \)
- **Narrow sense and primitive binary BCH code with \( \delta = 5 \)**: Let \( n = 2^m - 1 \) and \( \alpha \) primitive in GF(\( 2^m \)). With \( g(x) = p^{(1)}(x)p^{(3)}(x) \) we get \( \delta = 5 \). E.g., \( n = 15 \implies \)

\[
g(x) = (1 + x + x^4)(1 + x + x^2 + x^3 + x^4)
\]

For this code, \( n = 3 \cdot 5 \implies \text{d}_{\text{min}} = \delta = 5 \).
BCH Codes Cannot Achieve Capacity

- **Theorem**: There does not exist a sequence of \([n, k, d]\) primitive BCH codes over \(\text{GF}(q)\) with both \(d/n\) and \(k/n\) bounded away from zero as \(n \to \infty\).

Decoding Binary BCH Codes

- Let \(C\) be a narrow-sense and primitive \([n, k, d]\) BCH code over \(\text{GF}(2)\) of designed distance \(\delta = 2\tau + 1\).
- Let \(\alpha \in \text{GF}(2^m)\) be a primitive \(n\)th root of unity, with \(m\) the smallest integer such that \(n|2^m - 1\).
- Assume a codeword \(c = (c_0, \ldots, c_{n-1}) \in C\) is transmitted over a binary (memoryless) channel, resulting in
  \[y = (y_0, \ldots, y_{n-1}) = c + e\]
  with \(e = (e_0, \ldots, e_{n-1}) \in \text{GF}^n(2)\) of weight \(w\).
- Polynomials:
  \[c(x) = \sum_{m=0}^{n-1} c_m x^m, \quad y(x) = \sum_{m=0}^{n-1} y_m x^m, \quad e(x) = \sum_{m=0}^{n-1} e_m x^m\]
• The error locator polynomial $\Lambda(x)$: Assume that the non-zero components of $e$ are $e_{i_1}, \ldots, e_{i_w}$, and let

$$\Lambda(z) = \prod_{r=1}^{w} (1 - X_r z) = 1 + \sum_{r=1}^{w} \Lambda_r z^r$$

where $X_r = \alpha^{i_r}$ are the error locators

• Roots of $\Lambda(z)$ in $GF(2^m)$ known $\implies e$ known

• Decoding:
  1. Compute $A_i = y(\alpha^i)$, $i = 1, \ldots, \delta - 1$
  2. Find $\Lambda(z)$ from $A_1, \ldots, A_{\delta-1}$
  3. Compute the roots of $\Lambda(z) \rightarrow e(x)$

  • Will correct all errors of weight $w \leq \tau$
  • Polynomial (not exponential) complexity!

• Compute $A_i = y(\alpha^i)$, $i = 1, \ldots, \delta - 1$:
  • Divide $y(x)$ by the minimal polynomial $p(i)(x)$ of $\alpha^i$,
    
    $$y(x) = q(x)p(i)(x) + r(x),$$

    and set $x = \alpha^i$ in the remainder $r(x)$, $A_i = y(\alpha^i) = r(\alpha^i)$

• Equivalent to computing the syndrome: with $H$ on the form $H_{BCH}$ we get

$$s = Hy^T = He^T = \begin{bmatrix} y(\alpha) \\ y(\alpha^3) \\ \vdots \\ y(\alpha^{\delta-2}) \end{bmatrix} = \begin{bmatrix} e(\alpha) \\ e(\alpha^3) \\ \vdots \\ e(\alpha^{\delta-2}) \end{bmatrix} = \begin{bmatrix} A_1 \\ A_3 \\ \vdots \\ A_{\delta-2} \end{bmatrix}$$

and then we can get $A_2 = A_1^2$, $A_4 = A_2^2$, $A_{\delta-1} = A_{(\delta-1)/2}$.
• Compute \( \Lambda(z) \) from \( A_i, \ i = 1, \ldots, \delta - 1 \):
  - Newton’s identities (tailored to this problem):
    \[
    \begin{bmatrix}
      1 & 0 & 0 & 0 & 0 & \cdots & 0 \\
      A_2 & A_1 & 1 & 0 & 0 & \cdots & 0 \\
      A_4 & A_3 & A_2 & A_1 & 1 & \cdots & 0 \\
      \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
      A_{2w-4} & A_{2w-5} & \cdots & \cdots & A_{w-3} & \cdots & A_{w-1} \\
      A_{2w-2} & A_{2w-3} & \cdots & \cdots & A_{w-1}
    \end{bmatrix}
    \begin{bmatrix}
      \Lambda_1 \\
      \Lambda_2 \\
      \Lambda_3 \\
      \vdots \\
      \Lambda_{w-1} \\
      \Lambda_w
    \end{bmatrix}
    =
    \begin{bmatrix}
      A_1 \\
      A_3 \\
      A_5 \\
      \vdots \\
      A_{2w-3} \\
      A_{2w-1}
    \end{bmatrix}
    \]
  
  as long as \( w \leq \tau = (\delta - 1)/2 \)
  
  \( \{A_i\} \rightarrow \Lambda(z) \) not unique \( \implies \) choose \( \Lambda(z) \) of lowest degree

• Not feasible for large \( \tau \)'s \( \implies \) use instead the Berlekamp–Massey algorithm to find \( \Lambda(z) \)...

• Find the roots of \( \Lambda(z) \):
  - An error in coordinate \( i \) \( \iff \) \( \Lambda(\alpha^{-i}) = 0 \);
    - simply test \( \Lambda(\alpha^{-i}) = 0 \) for \( i = 1, \ldots, n \) (Chien search)

• Nonbinary BCH codes: Same principles apply, R6 describes the general approach...

• More than \( \tau \) errors: The method described only works for \( \leq \tau = (\delta - 1)/2 \) errors, i.e., full nearest neighbor decoding is not implemented;
  - Complete NN decoding algorithms (of polynomial complexity) known in many cases, but need often be tailored to specific codes...
  - The list decoding approach: see R9
  - Full search NN decoding always possible, but has exponential complexity...
Reed–Solomon Codes

- **Definition:** A Reed–Solomon (RS) code over $\text{GF}(q)$ is a BCH code of length $N = q - 1$, that is,
  \[ g(x) = (x - \alpha^b)(x - \alpha^{b+1}) \cdots (x - \alpha^{b+\delta-2}) \]
  for some $b \geq 0$ and $\delta \geq 2$, and with $\alpha$ primitive $\in \text{GF}(q)$
  - Zeros and symbols in the same field, $\text{GF}(q)$
  - Dimension $K = N - \delta + 1$
  - The Singleton bound $d_{\text{min}} \leq N - K + 1 
  \implies$  
    - $d_{\text{min}} = \delta$
    - maximum distance separable code

Encoding RS Codes

- **RS codes are cyclic:** Encode as (non-binary) cyclic codes...
  - **Alternative:** Assume an $[N, K]$ RS code, and let
    \[ u(x) = u_0 + u_1 x + \cdots + u_{K-1} x^{K-1} \]
  correspond to the message symbols $u_0, \ldots, u_{K-1} \in \text{GF}(q)$, then
  \[ c(x) = u(1) + u(\alpha)x + u(\alpha^2)x^2 + \cdots + u(\alpha^{N-1})x^{N-1} \]
  is a codeword.
Decoding RS Codes

- *RS codes are BCH codes*: Decode as non-binary BCH codes.
- Alternative *list* decoding: See R9.