# 7. Codes Related to GRS Codes

# Alternant Codes

- Let $\mathbb{F} = \mathbb{F}_q$ and let $\mathcal{C}_{\text{GRS}}$ be an $[N, K, D]$ GRS code over $\Phi = \mathbb{F}_{q^m}$. The set of codewords of $\mathcal{C}_{\text{GRS}}$ with coordinates in $\mathbb{F}$, is called an *alternant code*, $\mathcal{C}_{\text{alt}} = \mathcal{C}_{\text{GRS}} \cap \mathbb{F}^N$. For a PCM $H_{\text{GRS}}$ of $\mathcal{C}_{\text{GRS}}$, we have

$$\mathbf{c} \in \mathcal{C}_{\text{alt}} \quad \Longleftrightarrow \quad \mathbf{c} \in \mathbb{F}^N \text{ and } H_{\text{GRS}}\mathbf{c}^T = \mathbf{0}.$$

This is also called a *sub-field sub-code*.

$$H_{\text{GRS}} = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_N \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_N^2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{N-K-1} & \alpha_2^{N-K-1} & \ldots & \alpha_n^{N-K-1} \end{pmatrix} \begin{pmatrix} v_1 & & & \\ & v_2 & & 0 \\ 0 & & \ddots & \\ & & & v_N \end{pmatrix}.$$

# Alternant Codes

$$H_{\mathrm{GRS}} = \begin{pmatrix} v_1 & v_2 & \dots & v_N \\ v_1\alpha_1 & v_2\alpha_2 & \dots & v_n\alpha_N \\ v_1\alpha_1^2 & v_2\alpha_2^2 & \dots & v_n\alpha_N^2 \\ \vdots & \vdots & \vdots & \vdots \\ v_1\alpha_1^{N-K-1} & v_2\alpha_2^{N-K-1} & \dots & v_n\alpha_n^{N-K-1} \end{pmatrix}.$$

- Let $[n, k, d]$ be the parameters of $\mathcal{C}_{\mathrm{alt}}$. Clearly, $n = N$, and $d \geq D$; $D$ is called the *designed distance*.
  Each row of $H_{\mathrm{GRS}}$ translates to $\leq m$ independent rows over $\mathbb{F}$, so

  $$n - k \leq (N - K)m = (D - 1)m \quad \Longrightarrow \quad k \geq n - (D - 1)m$$

  Decoding: can be done with the same algorithm that decodes $\mathcal{C}_{\mathrm{GRS}}$.

# Binary Narrow-Sense Alternant Codes

- Consider $F = \mathbb{F}_2$ and $\mathcal{C}_{\text{GRS}}$ *narrow sense* $(v_j = \alpha_j)$ over $\mathbb{F}_{2^m}$, with *odd* $D$ and $N \leq 2^m - 1$.

$$H_{\text{GRS}} = \begin{pmatrix} \alpha_1 & \alpha_2 & \ldots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_n^2 \\ \alpha_1^3 & \alpha_2^3 & \ldots & \alpha_n^3 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{D-1} & \alpha_2^{D-1} & \ldots & \alpha_n^{D-1} \end{pmatrix}$$

For $\mathbf{c} \in \mathbb{F}_2^N$,

$$\mathbf{c} \in \mathcal{C}_{\text{alt}} \quad \Longleftrightarrow \quad \sum_{j=1}^{n} c_j \alpha_j^i = 0 \quad \text{for} \quad i = 1, 2, 3, \ldots, D{-}1 \, .$$

Over $\mathbb{F}_2$,

$$\sum_{j=1}^{n} c_j \alpha_j^i = 0 \quad \Longleftrightarrow \quad \sum_{j=1}^{n} c_j \alpha_j^{2i} = 0$$

Therefore, check equations for even values of $i$ are dependent, and the redundancy bound can be improved to

$$n - k \leq \frac{(D{-}1)m}{2} \, .$$

# Binary Narrow-Sense Alternant Codes

- A more compact PCM for binary narrow-sense $\mathcal{C}_{\mathrm{alt}}$:

$$\begin{pmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \alpha_1^3 & \alpha_2^3 & \dots & \alpha_n^3 \\ \alpha_1^5 & \alpha_2^5 & \dots & \alpha_n^5 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{D-2} & \alpha_2^{D-2} & \dots & \alpha_n^{D-2} \end{pmatrix}$$

- Decoding: same as $\mathcal{C}_{\mathrm{GRS}}$, but *error values* not needed
  $\Rightarrow$ simpler key equation algorithm.

# BCH Codes

- *Bose-Chaudhuri-Hocquenghem (BCH)* codes are alternant codes that correspond to conventional RS codes.

  For $\mathcal{C}_{\mathrm{RS}} : [N, K, D]$ over $\mathbb{F}_{q^m}$, we have $\mathcal{C}_{\mathrm{BCH}} = \mathbb{F}_q^N \cap \mathcal{C}_{\mathrm{RS}}$.

$$H_{\mathrm{RS}} = \begin{pmatrix} 1 & \alpha^b & \alpha^{2b} & \cdots & \alpha^{(N-1)b} \\ 1 & \alpha^{b+1} & \alpha^{2(b+1)} & \cdots & \alpha^{(N-1)(b+1)} \\ \vdots & \vdots & \vdots & \vdots & \\ 1 & \alpha^{b+D-2} & \alpha^{2(b+D-2)} & \cdots & \alpha^{(N-1)(b+D-2)} \end{pmatrix}$$

As before, when $b=1$, we can eliminate even-numbered rows

- As with RS codes, to obtain a *cyclic* code, we choose $N$ a divisor of $q^m - 1$. More often, we use a shortened code, where $N \leq q^m - 1$ is arbitrary. We lose the cyclic property, but all other properties hold.

# BCH Codes

For $\mathcal{C}_{\mathrm{RS}} : [N, K, D]$ over $\mathbb{F}_{q^m}$, $\mathcal{C}_{\mathrm{BCH}} = \mathbb{F}_q^N \cap \mathcal{C}_{\mathrm{RS}}$.

$$H_{\mathrm{RS}} = \begin{pmatrix} 1 & \alpha^b & \alpha^{2b} & \cdots & \alpha^{(N-1)b} \\ 1 & \alpha^{b+1} & \alpha^{2(b+1)} & \cdots & \alpha^{(N-1)(b+1)} \\ \vdots & \vdots & \vdots & \vdots & \\ 1 & \alpha^{b+D-2} & \alpha^{2(b+D-2)} & \cdots & \alpha^{(N-1)(b+D-2)} \end{pmatrix}$$

As before, when $b=1$, we can eliminate even-numbered rows

**Summary of BCH (and shortened BCH) code definition**

- Code of length $1 \le n \le q^m - 1$ over $\mathbb{F}_q$ for some choice of $m$. If we want a cyclic code, we pick $m$ to be the smallest integer such that $n | (q^m - 1)$.
- Let $\alpha \in \mathbb{F}_{q^m}$ be an element of order $n' \ge n$ ($n' = n$ for a cyclic code).
- $D > 0$, $b$: design parameters

$$\mathcal{C}_{\mathrm{BCH}} = \left\{ c(x) \in (\mathbb{F}_q)_n[x] \; : \; c(\alpha^\ell) = 0, \; \ell = b, b+1, \ldots, b+D-2 \right\}$$

- BCH codes are widely used in practice, for example, in *flash memories*.
- BCH codes are often superior to RS codes on the BSC.

# BCH Code Example

We design a BCH code of length $n = 15$ over $\mathbb{F}_2$ that can correct 3 errors. The code is primitive, of length $15$ with roots in $\mathbb{F}_{2^4}$.

- $m = 4$.
- $b = 1 \quad \Longrightarrow$ narrow-sense
- $D = 7 \quad \Longrightarrow$ 3-error correcting
- $n - k \leq (D-1)m/2 = 12$
- resulting $\mathcal{C}_{\mathrm{BCH}}$ is $[15, \geq 3, \geq 7]$ over $\mathbb{F}_2$
- Let $\alpha$ be a primitive element of $\Phi = \mathbb{F}_{2^4}$, which we choose as a root of $p(x) = x^4 + x + 1$ (primitive polynomial).
- a $12 \times 15$ *binary* PCM of the code can be obtained by representing the entries in $H_\Phi$ below as column vectors in $\mathbb{F}_2^4$.

$$H_\Phi = \begin{pmatrix} 1 & \alpha & \alpha^2 & \ldots & \alpha^j & \ldots & \alpha^{13} & \alpha^{14} \\ 1 & \alpha^3 & \alpha^6 & \ldots & \alpha^{3j} & \ldots & \alpha^{39} & \alpha^{42} \\ 1 & \alpha^5 & \alpha^{10} & \ldots & \alpha^{5j} & \ldots & \alpha^{65} & \alpha^{70} \end{pmatrix}$$

Notice that $\alpha^{15} = 1$, so $\alpha^{39} = \alpha^9$, etc.

# BCH Code Example (continued)

- A codeword $\mathbf{c} \in \mathcal{C}_{\mathrm{BCH}}$ satisfies $c(\alpha) = 0$. Therefore,

$$0 = c(\alpha)^2 = \left(\sum_{i=0}^{n-1} c_i x^i\right)^2 = \sum_{i=0}^{n-1} c_i^2 x^{2i} = \sum_{i=0}^{n-1} c_i x^{2i} = c(\alpha^2).$$

  For the same reason, $c(\alpha) = c(\alpha^2) = c(\alpha^4) = c(\alpha^8) = 0$
  $\Rightarrow\ M_1(x)$, the minimal polynomial of $\alpha$, divides $c(x)$.
- Similarly for $M_3(x)$ and $M_5(x)$, the min. polys. of $\alpha^3$ and $\alpha^5$ resp.
- Let $g(x) = M_1(x)M_3(x)M_5(x)$. Then,

$$\mathbf{c} \in \mathcal{C}_{\mathrm{BCH}} \quad \Leftrightarrow \quad g(x)|c(x).$$

- $g(x)$ is the *generator polynomial of* $\mathcal{C}_{\mathrm{BCH}}$, which is presented as a (shortened) *cyclic binary code*.
- In the example,
  $M_1(x) = x^4 + x + 1$,
  $M_3(x) = x^4 + x^3 + x^2 + x + 1$,
  $M_5(x) = x^2 + x + 1$.

$$\Rightarrow \quad g(x) = x^{10} + x^8 + x^5 + x^4 + x^2 + x + 1\,.$$

# BCH Code Example (continued)

- As with RS codes, we have the polynomial (cyclic) interpretation of BCH codes: $u(x) \mapsto c(x){=}u(x)g(x)$, with $u(x) \in \mathbb{F}_2[x]$ (a binary polynomial of degree $< k$), corresponding to a non-systematic *binary* generator matrix

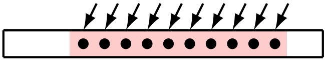
$$G = \begin{pmatrix} g_0 & g_1 & \dots & g_{n-k} & & & & \\ & g_0 & g_1 & \dots & g_{n-k} & & \mathbf{0} & \\ \mathbf{0} & & \ddots & \ddots & \dots & \ddots & & \\ & & & g_0 & g_1 & \dots & g_{n-k} \end{pmatrix} \qquad (g_{n-k}{=}1,\ k \text{ rows})$$

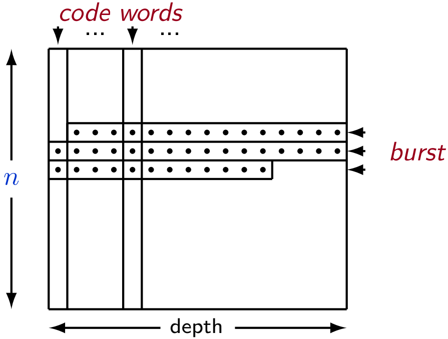
- In the example, this representation also implies that $k_{BCH} = 15 - 10 = 5$, the rank of $G$.
- Codes with dimension better than the bound are obtained when some of the minimal polynomials $M_i$ are of degree less than $m$.
  This happened, in our example, for $M_5$.
- As in the RS case, we can construct a *systematic encoder* based on $g(x)$ and using a *binary* feedback shift-register.

*The $[15, 5, 7]$ BCH code in the example is used for format information in QR codes.*

# Interleaving and Burst Error Correction

- *Burst errors*

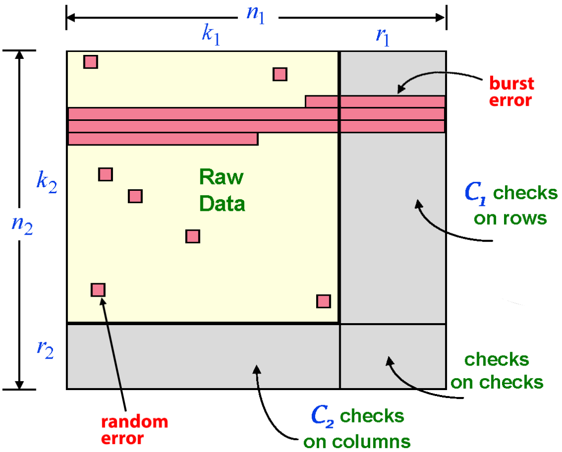




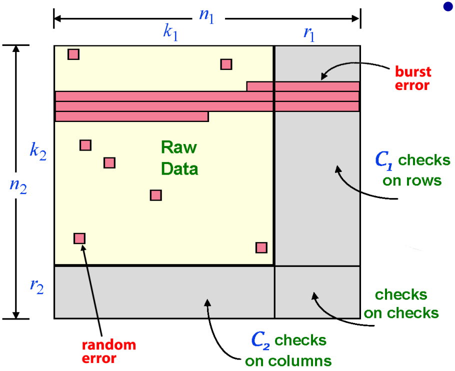

- *Interleaving* spreads bursts of errors among codewords, so that each codeword is affected by a small number of errors.
- Cost: increased *latency*

# Product codes

Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be $[n_1, k_1, d_1]$ and $[n_2, k_2, d_2]$ (usually RS) codes, resp.



# Decoding product codes




- A decoding strategy:
  - Use a (small) part of the $\mathcal{C}_1$ redundancy to correct random errors, and the rest for robust error detection (so that burst errors in rows will be detected with high probability).
  - Mark detected corrupted rows as *erased*.
  - Use the column code $\mathcal{C}_2$ to correct the erasures (and remaining random errors, if any, and if possible). Recall that erasures are "cheaper" to correct than full errors.
  - Other strategies are possible, including row/column iterations.

# Concatenated Codes

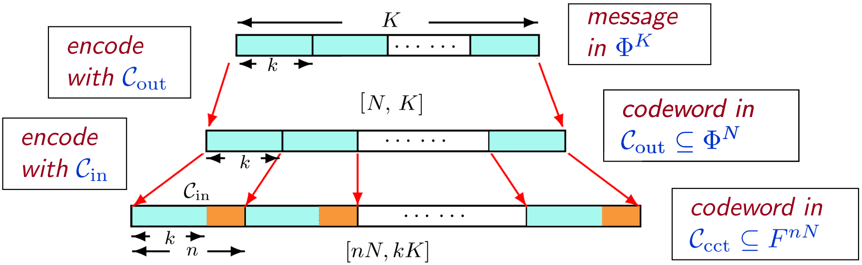
- Let $\mathbb{F} = \mathbb{F}_q$ and $\Phi = \mathbb{F}_{q^k}$, $k > 1$.
- Let $\mathcal{C}_{\text{out}}$ be an $[N, K, D]$ code over $\Phi$ (the *outer code*).
- Let $\mathcal{C}_{\text{in}}$ be an $[n, k, d]$ code over $\mathbb{F}$ (the *inner code*).
  - Notice that the *dimension* $k$ of $\mathcal{C}_{\text{in}}$ is the same as the *extension degree* of $\Phi$ over $\mathbb{F}$.
- Represent $\Phi$ as vectors in $\mathbb{F}^k$ using a fixed basis of $\Phi$ over $\mathbb{F}$.
- A *concatenated code* $\mathcal{C}_{\text{cct}}$ is defined by the following

**Encoding Procedure:**

**Input:** A *message* $\mathbf{u}$ of length $K$ over $\Phi$.
**Output:** A *codeword* $\mathbf{c}_{\text{cct}}$ of length $nN$ over $\mathbb{F}$.

- **Step 1:** Encode $\mathbf{u}$ into a codeword $\mathbf{c}_{\text{out}} \in \mathcal{C}_{\text{out}}$.
- **Step 2:** Interpret each of the $N$ symbols of $\mathbf{c}_{\text{out}}$ as a word of length $k$ over $\mathbb{F}$. Encode it with $\mathcal{C}_{\text{in}}$.

# Concatenated Codes




- $\mathcal{C}_{\text{cct}}$ has parameters $[n_{\text{cct}}, k_{\text{cct}}, d_{\text{cct}}] = [nN, kK, \geq dD]$ over $F$.
- As with product codes, different decoding strategies are possible.
  - Typically, we use $\mathcal{C}_{\text{in}}$ for combined error correction/detection. When errors are detected without correction, the symbol is marked as *erased* for $\mathcal{C}_{\text{out}}$.
  - Then we use $\mathcal{C}_{\text{out}}$ to correct erasures and errors. The process may be iterative.
  - Forney's *Generalized Minimum Distance* decoding can correct up to $(dD-1)/2$ errors.

# Concatenated Codes

- $\mathcal{C}_{\text{out}}$ is typically taken to be a GRS code.
    - By letting $k$ grow, we can obtain arbitrarily long codes over $\mathbb{F}_q$, for fixed $q$.
    - By careful choice of $\mathcal{C}_{\text{in}}$, *very good codes* can be constructed this way.
        - Codes with $R_{\text{cct}}$ and $d_{\text{cct}}/n_{\text{cct}}$ bounded away from zero as $k \to \infty$, which can be constructed *explicitly* and have efficient encoding/decoding algorithms.
        - Even better, codes that *achieve channel capacity for the QSC channel*, still with explicit constructions and efficient encoding/decoding algorithms.
    - Variant: use a different $\mathcal{C}_{\text{in}}$ for each coordinate of $\mathcal{C}_{\text{out}}$.
    - Notice that what is exponential in $k$ may be linear in $N$: *ML decoding for $\mathcal{C}_{\text{in}}$ may be affordable*.