ELG 5372 Error Control Coding

Lecture 16: Decoding of BCH and RS Codes

uOttawa

L'Université canadienne
Canada's university

Université d'Ottawa | University of Ottawa

uOttawa.ca

Algebraic Decoding of BCH and RS Codes

- The algebraic decoding of BCH and RS codes has the following general steps:
  - Computation of the syndrome
  - Determination of an error location polynomial. The roots of this polynomial provide the location of the errors. There are many algorithms for finding this polynomial (Peterson's, Berlekamp-Massey, Peterson-Gorenstein-Zierler etc)
  - Determination of roots of error locator polynomial. Usally done by Chien search
  - For non-binary BCH and RS codes, error values must be found (usually using Forney's algorithm).

Computation of Syndrome

- For all examples, we will assume narrow-sense BCH or RS codes.
- We know that α, α^2, …, α^{2t} are roots of $g(x)$, therefore they are roots of $c(x)$ as well.
- Therefore $c(\alpha) = c(\alpha^2) = \ldots = c(\alpha^{2t})$.
- The received polynomial $r(x) = c(x)+e(x)$.
- Let $S_j = r(\alpha^j) = c(\alpha^j)+e(\alpha^j) = e(\alpha^j)$ for $j = 1, 2, \ldots, 2t$.
- The values S_1, S_2, …, S_{2t} are the syndromes of the received polynomial.

Computation of Syndrome

$$S_j = \sum_{i=1}^{n} e_i \left(\alpha^j\right)^i = \sum_{i=1}^{n} e_i \alpha^{ij} \qquad (*)$$

Suppose that $r(x)$ has v errors in it and that they are in positions $i_1, i_2, \ldots, i_v$. Then (*) becomes:

$$S_j = \sum_{l=1}^{v} e_{i_l} \left(\alpha^j\right)^{i_l} = \sum_{l=1}^{v} e_{i_l} \left(\alpha^{i_l}\right)^j = \sum_{l=1}^{v} e_{i_l} X_l^j \qquad (**)$$

where $X_l = \alpha^{i_l}$ and $j = 1, 2, \ldots, 2t$

Computation of Syndrome for Binary Codes

- For binary codes $e_{il} = 1$. Therefore (**) becomes

$$S_j = \sum_{l=1}^{v} X_l^j$$

$$\text{where } X_l = \alpha^{i_l} \text{ and } j = 1, 2, \ldots, 2t$$

- If we know X_l, then we know the location of the error.
  - For example, if $X_1 = \alpha^2$, then by definition, $i_1 = 2$ and the error is in digit r_2.

The Error Locator Polynomial for Binary BCH Codes

- We obtain the following set of equations:

$$S_1 = X_1 + X_2 + \ldots + X_v$$

$$S_2 = X_1^2 + X_2^2 + \ldots + X_v^2$$

$$\vdots$$

$$S_{2t} = X_1^{2t} + X_2^{2t} + \ldots + X_v^{2t}$$

- The equations are said to be power-sum symmetric functions and it gives us a set of 2*t* equations with *v* unknowns.

The Error Locator Polynomial for Binary BCH Codes

- The set of power symmetric functions is a solvable set of functions (for $v \le t$). However, it is computationally complex.
- Therefore a new polynomial is introduced. This is the error locator polynomial:

$$\Lambda(x) = \prod_{l=1}^{v}(1 - X_l x) = \Lambda_v x^v + \Lambda_{v-1} x^{v-1} + \ldots + \Lambda_1 x + 1$$

- X_l^{-1} is a root of this polynomial.

Finding the Error Locator Polynomial

Let us consider the case when $v = 2$.

$$\Lambda(x) = (1 - X_1 x)(1 - X_2 x) = 1 - (X_1 + X_2)x + X_1 X_2 x^2$$

$\Lambda_1 = -(X_1+X_2)$ and $\Lambda_2 = X_1X_2$

We can see that $S_1+\Lambda_1 = 0$

$S_2 = X_1^2+X_2^2$, $S_2+2\Lambda^2 = (X_1^2+2X_1X_2+X_2^2) = (X_1+X_2)^2$

$S_2+S_1\Lambda_1+2\Lambda_2 = 0$

Also $S_3+\Lambda_1S_2+\Lambda_2S_1 = 0$

And $S_4+\Lambda_1S_3+\Lambda_2S_2 = 0$

Finding the Error Locator Polynomial 2

- We can extend this to arbitrary *v*:

$$
\begin{array}{lcl}
k=1 & : & S_1+\Lambda_1=0 \\
k=2 & : & S_2+S_1\Lambda_1+2\Lambda_2=0 \\
\vdots & & \\
k=v & : & S_v+S_{v-1}\Lambda_1+S_{v-2}\Lambda_2+\ldots+v\Lambda_v=0 \\
k=v+1 & : & S_{v+1}+S_v\Lambda_1+S_{v-1}\Lambda_2+\ldots+S_1\Lambda_v=0 \\
k=v+2 & : & S_{v+2}+S_{v+1}\Lambda_1+S_v\Lambda_2+\ldots+S_2\Lambda_v=0 \\
\vdots & & \\
k=2t & & S_{2t}+S_{2t-1}\Lambda_1+S_{v-1}\Lambda_2+\ldots+S_{2t-v}\Lambda_v=0
\end{array}
$$

These are Newton's identities

Finding the Error Locator Polynomial 3

Let $v = t$

$$\begin{bmatrix} S_1 & S_2 & S_3 & \cdots & S_v \\ S_2 & S_3 & S_4 & \cdots & S_{v+1} \\ S_3 & S_4 & S_5 & \cdots & S_{v+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ S_v & S_{v+1} & S_{v+2} & \cdots & S_{2v-1} \end{bmatrix} \begin{bmatrix} \Lambda_v \\ \Lambda_{v-1} \\ \Lambda_{v-2} \\ \vdots \\ \Lambda_1 \end{bmatrix} = -\begin{bmatrix} S_{v+1} \\ S_{v+2} \\ S_{v+3} \\ \vdots \\ S_{2v} \end{bmatrix}$$

$$\mathbf{M}_v \boldsymbol{\Lambda} = -\mathbf{S}$$

Peterson-Gorenstein-Zierler Algorithm

- Set $v = t$
- Form $\mathbf{M}_v$ and determine if $\mathbf{M}_v$ is invertible (compute det($\mathbf{M}_v$), if det($\mathbf{M}_v$) = 0, $\mathbf{M}_v$ is not invertible.
  - If not invertible, it means there are less than t errors
  - Set $v = t$-1 and repeat step
- Once $\mathbf{M_v}$ is invertible, compute $\Lambda = \mathbf{M}_v^{-1}(-\mathbf{S})$

Example

- Consider the binary (15,7) BCH code.
- This is a two error correcting code.
- Suppose $r(x) = x^7$.
- $S_1 = \alpha^7$, $S_2 = \alpha^{14}$, $S_3 = \alpha^6$, $S_4 = \alpha^{13}$,
- Assume $v = 2$

$$\mathbf{M}_2 = \begin{bmatrix} \alpha^7 & \alpha^{14} \\ \alpha^{14} & \alpha^6 \end{bmatrix}, \ \det(\mathbf{M}_2) = \alpha^{13} - \alpha^{13} = 0$$

Example cont'd

- Therefore we assume that $v = 1$
- $\mathbf{M}_1 = [\alpha^7]$
- Then $a^7\Lambda_1 = -\alpha^{14}$
- Or $\Lambda_1 = -\alpha^7$.
- The error locator polynomial is $\Lambda(x) = 1-\alpha^7 x$ (or $1+\alpha^7 x$). This has root $x = \alpha^8$. Therefore $X_1^{-1} = \alpha^8$, or $X_1 = \alpha^7$. Error position is r_7 in $r(x)$. Therefore $c(x) = r(x)-x^7 = 0$.

Example 2

- For the same code, assume that $r(x) = x^2+x^5$.
- $S_1 = \alpha^2+\alpha^5 = \alpha$, $S_2 = \alpha^4+\alpha^{10} = \alpha^2$, $S_3 = \alpha^6+1 = \alpha^{13}$ and $S_4 = \alpha^8+\alpha^5 = \alpha^4$.

$$\mathbf{M}_2 = \begin{bmatrix} \alpha & \alpha^2 \\ \alpha^2 & \alpha^{13} \end{bmatrix}, \det(\mathbf{M}_2) = \alpha^{14} - \alpha^4 = \alpha^9$$

$$\mathbf{M}_2^{-1} = \frac{1}{\alpha^9}\begin{bmatrix} \alpha^{13} & \alpha^2 \\ \alpha^2 & \alpha \end{bmatrix} = \begin{bmatrix} \alpha^4 & \alpha^8 \\ \alpha^8 & \alpha^7 \end{bmatrix}$$

$$\Lambda = \begin{bmatrix} \Lambda_2 \\ \Lambda_1 \end{bmatrix} = \begin{bmatrix} \alpha^4 & \alpha^8 \\ \alpha^8 & \alpha^7 \end{bmatrix}\begin{bmatrix} \alpha^{13} \\ \alpha^4 \end{bmatrix} = \begin{bmatrix} \alpha^7 \\ \alpha \end{bmatrix}$$

Example 2 cont'd

- Therefore $\Lambda(x) = \alpha^7 x^2+\alpha x+1 = (\alpha^2 x+1)(\alpha^5 x+1)$
- The roots are $X_1^{-1} = \alpha^{13}$ and $X_2^{-1} = \alpha^{10}$. Therefore $X_1 = a^2$ and $X_2 = \alpha^5$.
- This means r_2 and r_5 are incorrect.
- $c(x) = r(x)+x^2+x^5 = 0$.

Simplifications for Binary Codes

- For GF(2^m), $(X+Y)^2 = (X^2+Y^2)$.
- Therefore $S_{2j} = S_j^2$.
- Also $nX = 0$ if n is even and $nX = X$ if n is odd.

Newton's Identities

$$\begin{array}{lcc}
k=1 & : & S_1+\Lambda_1=0 \\
k=2 & : & S_2+S_1\Lambda_1+2\Lambda_2=0 \\
\vdots & & \\
k=v & : & S_v+S_{v-1}\Lambda_1+S_{v-2}\Lambda_2+\ldots+v\Lambda_v=0 \\
k=v+1 & : & S_{v+1}+S_v\Lambda_1+S_{v-1}\Lambda_2+\ldots+S_1\Lambda_v=0 \\
k=v+2 & : & S_{v+2}+S_{v+1}\Lambda_1+S_v\Lambda_2+\ldots+S_2\Lambda_v=0 \\
\vdots & & \\
k=2t & & S_{2t}+S_{2t-1}\Lambda_1+S_{v-1}\Lambda_2+\ldots+S_{2t-v}\Lambda_v=0
\end{array}$$

All the even equations are redundant

Newton's identities minus redundant equations

$$
\begin{aligned}
k &= 1 &&: & S_1 + \Lambda_1 &= 0 \\
k &= 3 &&: & S_3 + S_2\Lambda_1 + S_1\Lambda_2 + \Lambda_3 &= 0 \\
&\vdots \\
k &= 2t-1 &&: & S_{2t-1} + S_{2t-2}\Lambda_1 + \ldots + S_{t-1}\Lambda_t &= 0
\end{aligned}
$$

Newton's identities minus redundant equations in matrix form

- **AΛ = -S**

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ S_2 & S_1 & 1 & 0 & \cdots & 0 & 0 \\ S_4 & S_3 & S_2 & S_1 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ S_{2t-4} & S_{2t-5} & S_{2t-6} & S_{2t-7} & \cdots & S_{t-2} & S_{t-3} \\ S_{2t-2} & S_{2t-3} & S_{2t-4} & S_{2t-5} & \cdots & S_t & S_{t-1} \end{bmatrix} \begin{bmatrix} \Lambda_1 \\ \Lambda_2 \\ \vdots \\ \Lambda_t \end{bmatrix} = -\begin{bmatrix} S_1 \\ S_3 \\ \vdots \\ S_{2t-1} \end{bmatrix}$$

Peterson Algorithm

- Assume there are t errors. If there are in fact t errors, $\mathbf{A}$ is invertible.
  - If $\mathbf{A}$ not invertible, delete last two rows and last two columns and repeat
- Once $\mathbf{A}$ is invertible, $\Lambda = \mathbf{A}^{-1}(-\mathbf{S})$.

Coefficients for Error Locator Polynomial for small number of errors

- Using Peterson's algorithm, explicit expressions for Λ_i have been computed for codes that can correct a small number of errors.
- 1 error correcting, $\Lambda_1 = S_1$
- 2 error correcting, $\Lambda_1 = S_1$ and $\Lambda_2 = (S_3+S_1^3)/S_1$.
- 3 error correcting, $\Lambda_1 = S_1$ and $\Lambda_2 = (S_1^2S_3+S_5)/(S_1^3+S_3)$, $\Lambda_3 = (S_1^3+S_3)+S_1\Lambda_2$.
- Others can be found on page 252 of text.