Error Detection & Correction

Encode in.data.frame k bits out.data.frame n bits

n > k
r = n - k = redundant bits added to detect errors
Error Detection

- Example: $n = 2k$
  - Repeat the bits
  - $k$ redundant bits

\[
\text{Redundancy Ratio} = \frac{n}{k}
\]

Error Detection

- Example: Simple Parity Check
- Even Parity
  \(\text{number of ‘}1\text{’ per block is even}\)
- Odd Parity
  \(\text{number of ‘}1\text{’ per block is odd}\)
- Redundancy ratio = $k+1/k$
Error Detection

- Performance criteria for error detection codes is the probability of an undetected error, $P_e$.

Let $p = \text{probability of a bit error}$ and assume random errors.

then

$\text{Prob[1 bit error in a block of } n \text{ bits]} = P(1) = np(1-p)^{n-1}$

for $m$ errors

$P(m) = \frac{n!}{(n-m)!m!}p^m(1-p)^{n-m}$

---

Example: Find the time between undetected errors for:

- $n = 1000$
- $p = 1$ in million
- Rate = 100 Mb/s

For simple parity check:

$P_e = P(2) + P(4) + P(6) + \ldots$

Note $P(m) \approx 0$ for $M > 3$.

$P_e = \frac{n(n-1)}{n}p^2(1-p)^{n-2}$

Example: Find the time between undetected errors for:

- $n = 1000$
- $p = 1$ in million
- Rate = 100 Mb/s

$P_e = 5 \times 10^{-7}$

Time to transmit a block = 10$\mu$s

Time between undetected errors = $\frac{10\mu s}{5 \times 10^{-7}} = 20$ seconds
Block Codes

Representation and manipulation of message blocks

\( k \)-bit message is represented by a binary vector of length \( k \)

\[ \vec{d} = (d_1, d_2, d_3, \ldots, d_k) \]

Example, \( 01001100 \rightarrow (0,1,0,0,1,1,0,0) \)

Block Codes

\( n \)-bit code word is represented by a binary vector of length \( n \)

\[ \vec{c} = (c_1, c_2, c_3, \ldots, c_n) \]

Example,

\( 01001100110 \rightarrow (0,1,0,0,1,1,0,0,1,1,0) \)

Error Control Coding: Map \( k \)-bit message into \( n \)-bit code word
One Example Mapping

Let \( c_1 = d_1, \ c_2 = d_2, \ c_3 = d_3, \ldots \ c_k = d_k \)
\[ c_{k+1} = p_{11}d_1 \oplus p_{21}d_2 \oplus p_{31}d_3 \oplus p_{41}d_4 \ldots \oplus p_{k1}d_k \]
\[ c_{k+2} = p_{12}d_1 \oplus p_{22}d_2 \oplus p_{32}d_3 \oplus p_{42}d_4 \ldots \oplus p_{k2}d_k \]
\[ \vdots \]
\[ c_n = p_{1n}d_1 \oplus p_{2n}d_2 \oplus p_{3n}d_3 \oplus p_{4n}d_4 \ldots \oplus p_{kn}d_k \]

Note \( 0 \oplus 0 = 0, 1 \oplus 0 = 1, 0 \oplus 1 = 1, 1 \oplus 1 = 0 \)

---

Block codes: Example

- \( c_1 = d_1 \)
- \( c_2 = d_2 \)
- \( c_3 = d_3 \)
- \( c_4 = d_1 + d_3 \)
- \( c_5 = d_2 + d_3 \)
- \( c_6 = d_1 + d_2 \)
- Let \( d = (0, 0, 1) \)
  \[ C = (0, 0, 1, 1, 1, 0) \]
  Number of valid code words = \( 2^k \)
  Here \( k = 3 \) so 8 valid code words
Block Codes

In matrix form \( \bar{c} = \bar{d}G \)

where

\[
\begin{align*}
1 & 0 & 0 & 0 & 0 & p_{11} & p_{12} & p_{13} & p_{14} & p_{15} & p_{1r} \\
0 & 1 & 0 & 0 & 0 & p_{21} & p_{22} & p_{23} & p_{24} & p_{25} & p_{2r} \\
G &= 0 & 0 & 1 & 0 & 0 & p_{31} & p_{32} & p_{33} & p_{34} & p_{35} & p_{3r} \\
0 & 0 & 0 & 1 & 0 & p_{41} & p_{42} & p_{43} & p_{44} & p_{45} & p_{4r} \\
0 & 0 & 0 & 0 & 1 & p_{k1} & p_{k2} & p_{k3} & p_{k4} & p_{k5} & p_{kr}
\end{align*}
\]

\( G = [I_k, P] \) where \( I_k = k \times k \) identity matrix

\( G = \text{Code Generator Matrix} \)

---

Block Codes

- \( G \) Defines a Linear Systematic \((n,k)\) Code

<table>
<thead>
<tr>
<th>k Message Bits</th>
<th>r Code Bits</th>
</tr>
</thead>
</table>

Code Word of \( n \) bits
Block Codes: Example

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \quad P = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 \end{bmatrix}$$

There are 8 unique code words

$$c_1 = d_1G = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$c_2 = d_2G = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Problem: given a received message determine if an error has occurred.

Model for the received message:

$$\bar{r} = \bar{c} \oplus \bar{e}$$

where

$$\bar{e} = \text{error vector}$$

$$e_i = 1 \text{ means that a error occurred in the } i^{th} \text{ bit of the code word}$$

$$e = (001010)$$

Errors in: bit 3 & 5
Block Codes

- Define a *Parity Check Matrix* $H$

$$H = [P^t I_r] = \begin{bmatrix} p_{11} & \ldots & p_{k1} & 1 & 0 & 0 & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & 0 & 1 & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & 0 & 1 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & 0 & 1 \\ p_{1r} & \ldots & p_{kr} & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Block codes

For all $2^k$ valid code words, it can be shown that $\bar{c}H^t = 0$

Then $\bar{r}H^t = (\bar{c} \oplus \bar{e})H^t = \bar{c}H^t \oplus \bar{e}H^t = \bar{e}H^t$

and if $\bar{e}H^t \neq 0$ then $\bar{e} \neq 0$ and an error has occured.

So if $\bar{r}H^t \neq 0$ then an error has occured.

$\bar{s} = \bar{r}H^t = \text{Error Syndrome}$

Therefore, if $\bar{s} = 0$ then no error.
**Block Codes: Example**

\[
\begin{align*}
G &= \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \ 0 & 1 & 0 & 1 & 0 \ 0 & 0 & 1 & 1 & 1 \ 0 & 0 & 1 & 1 & 0 \ 0 & 0 & 1 & 0 & 1 \ 0 & 0 & 0 & 1 & 0 \ \end{bmatrix}, \\
P &= \begin{bmatrix} 0 & 1 & 1 \ 1 & 0 & 1 \ 1 & 1 & 0 \ \end{bmatrix}, \\
H &= \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \ 1 & 0 & 1 & 0 & 1 \ 1 & 1 & 0 & 0 & 0 \ \end{bmatrix}, \\
H^t &= \begin{bmatrix} 0 & 1 & 1 \ 1 & 0 & 1 \ 1 & 1 & 0 \ 1 & 0 & 0 \ 0 & 1 & 0 \ 0 & 0 & 1 \ \end{bmatrix}, \\
\tilde{r} &= \begin{bmatrix} 1 & 1 & 1 & 1 \ \end{bmatrix}, \\
\tilde{s} &= \begin{bmatrix} 1 & 1 & 1 & 1 \ \end{bmatrix} - \begin{bmatrix} 0 & 1 & 1 \ 1 & 0 & 1 \ 1 & 1 & 0 \ \end{bmatrix} = 0
\end{align*}
\]

An error has been detected.

**Hamming Distance**

A Geometric Interpretation of Error correcting Codes

- Hamming distance between two code words is defined as the number of differing bit positions.

Let

\[
c_i = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 \ \end{bmatrix}
\]

and

\[
c_j = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 \ \end{bmatrix}
\]

then the Hamming distance between \(c_i\) and \(c_j\) is \(d_{ij} = 2\).

Note \(d_{ij} = \sum c_i \oplus c_j\).

The minimum distance decoder selects the code word 'closest' in Hamming distance to the received message.

A minimum distance decoder can correct \(m\) errors where \(m\) is the largest integer not to exceed \(1/2(d_{\text{min}} - 1)\), where \(d_{\text{min}}\) is the minimum hamming distance between any two valid code words.
Binary Cyclic Codes

- A subclass of linear block codes are the binary cyclic codes.
- If $c_k$ is a code word in a binary cyclic code then a lateral (or cyclic) shift of it is also a code word.

If $\begin{bmatrix} c_1 & c_2 & c_3 & c_4 & c_5 \end{bmatrix}$ is a code word
then $\begin{bmatrix} c_5 & c_1 & c_2 & c_3 & c_4 \end{bmatrix}$ and
$\begin{bmatrix} c_4 & c_5 & c_1 & c_2 & c_3 \end{bmatrix}$ can also be code words.
Binary Cyclic Codes

Advantages:
- Ease of syndrome calculation
- Simple and efficient coding/decoding structure

Cyclic codes restrict the form of the G matrix

The cyclic nature of these codes indicates that there is a underlying pattern.

Generator Polynomial $g(x)$

$g(x) = 1x^r \oplus g_{r-1} \oplus g_{r-3}x^{r-1} \oplus g_{r-3}x^{r-2} \ldots \oplus 1$

Example:

$g(x) = x^{16} \oplus x^{15} \oplus x^2 \oplus 1$

or $g(x) = x^{16} + x^{15} + x^2 + 1$

$g_i = 0, 1$
Binary Cyclic Codes:
Construction of $G$ from $g(x)$

- 1) Use $g(x)$ to form the $k^{th}$ row of $G$.
- 2) Use $k^{th}$ row to form $(k-1)$ row by a shift left, or $xg(x)$.
- 3) If $(k-1)$ row in not in standard form then add $k^{th}$ row to shifted row, i.e., $(k-1)$ row becomes $xg(x) + g(x)$.
- 4) Continue to form rows from the row below.
- 5) If the $(k-j)$ row not in standard form the add $g(x)$ to the shifted row.

Example:

$g(x) = x^3 + 0x^2 + x + 1$

$n = 7, k = 4, r = 3$

**Step 1:** $k^{th}$ row of $G$ → \[
\begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 1 & 1
\end{bmatrix}
\]

**Step 2:** $k - 1$ row of $G$ → \[
\begin{bmatrix}
0 & 0 & 1 & 0 & 1 & 1 & 0
\end{bmatrix}
\]

Form: ok.

**Step 3:** $k - 2$ row of $G$ → \[
\begin{bmatrix}
0 & 1 & 0 & 1 & 1 & 0 & 0
\end{bmatrix}
\]

Form: not ok.

Add:
\[
\begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 1 & 1
\end{bmatrix}
\]
\[
\begin{bmatrix}
0 & 1 & 0 & 0 & 1 & 1 & 1
\end{bmatrix}
\]

Form: ok.

**Step 4:** $k - 3$ row of $G$ → \[
\begin{bmatrix}
1 & 0 & 0 & 1 & 1 & 1 & 0
\end{bmatrix}
\]

Form: not ok.

Add:
\[
\begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 1 & 1
\end{bmatrix}
\]
\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 0 & 1
\end{bmatrix}
\]

Form: ok.

---

$G = \begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 1
\end{bmatrix}$
Binary Cyclic Codes

Codes for this example are:

\[ \bar{c}_1 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \]

\[ \bar{c}_2 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \]

Note that \( \bar{c}_1 \) and \( \bar{c}_2 \) are cyclic shifts of each other.

Binary Cyclic Codes: Standard generator polynomials

**CRC-16** \( \rightarrow g(x) = x^{16} + x^{15} + x^2 + 1 \)

**CRC-CCITT** \( \rightarrow g(x) = x^{16} + x^{12} + x^5 + 1 \)

**CRC-12** \( \rightarrow g(x) = x^{12} + x^{11} + x^3 + x^2 + x + 1 \)

**AAL3/4** \( \rightarrow g(x) = x^{10} + x^9 + x^5 + x^4 + x + 1 \)

**AAL5** \( \rightarrow x^{32} + x^{26} + x^{22} + x^{16} + x^{11} + x^{10} + x^8 + x^7 + x^4 + x^2 + x + 1 \)
Interleaving

- Converts burst errors to random errors
- Impacts delay
- Used in CD players
  - Specifications: correct 90 ms of errors
  - at 44 kb/s & 16 bits/sample
    - 4000 consecutive bit errors