NETWORK PROTOCOLS AND SECURITY

Dr. G. Omprakash

Assistant Professor, ECE, KLEF

Aim of the session

To discuss error correction methods used in networks

Learning Outcomes

At the end of this session, you should be able to:

- Understand the concept of error correction
- Able to apply error Correction methods

Overview

Error Correction

Hamming Distance

- A frame consists of k data (i.e., message) bits and r redundant (i.e., check) bits
- The number of bit positions in which two codewords differ is called the **Hamming distance**
- The Hamming distance can easily be found if we apply the XOR operation on the two words and count the number of 1s in the result
- The Hamming distance $d(000, 011)$ is 2
- The Hamming distance $d(10101, 11110)$ is 3

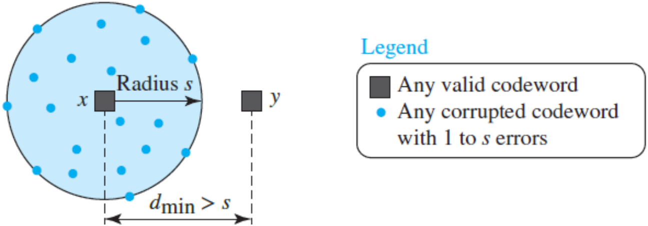
Min Hamming Distance for error detection

Datawords	*Codewords*	*Datawords*	*Codewords*
00	000	10	101
01	011	11	110

- Observe: The Hamming distance between any two codewords in the table is 2
- Sender encodes the dataword 01 as 011 and sends it to the receiver
  - Receiver receives 011 $\Longrightarrow$ valid codeword. Extract 01
  - Receives 111 (one-bit corrupted) $\Longrightarrow$ not a valid codeword, so discarded.
  - 000 (two-bits are corrupted) is received, valid codeword $\Longrightarrow$ extract 00
  - Hamming distance 2 $\Longrightarrow$ upto 1 error can only be detected !!

Min Hamming Distance

To guarantee the **detection** of up to s errors in all cases, the minimum Hamming distance in a block code must be $\mathbf{d_{min} = s + 1}$.



To guarantee the **correction** of up to s errors in all cases, the minimum Hamming distance in a block code must be $\mathbf{d_{min} = 2s + 1}$.

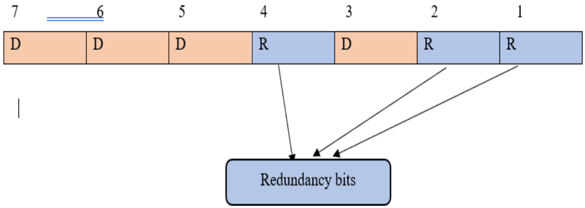
Question: Given k, what is the lower limit on the number of check bits (r) needed to correct single errors?

$$(k + r + 1) \leq 2^r$$

Number of data bits (m)	Number of redundancy bits (r)	Total bits (m+r)
1	2	3
2	3	5
3	3	6
4	3	7
5	4	9
6	4	10
7	4	11

Hamming code:

- Check bit positions: 1,2,4,8,16..
- Data bit positions: 3,5,6,7,9,...



r1-bit parity

Out of three binary bits we select positions with 1 in Least significant bit for computing r_1 i.e 1,3,5,7

7(111)	6(110)	5(101)	4(100)	3(011)	2(010)	1(001)
D	D	D	r4	D	r2	r1
1	1	0		1		0

r1-----1,3,5,7

_, 1, 0, 1-----→ must satisfy even parity as per the assumption. As the count of 3,5,7 position is giving even number of 1's the r1=0

r2-bit parity

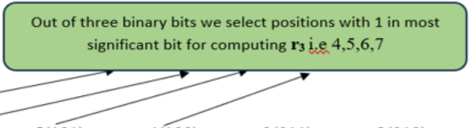
Out of three binary bits we select positions with 1 in middle bit for computing r_2 i.e 2,3,6,7

7(111)	6(110)	5(101)	4(100)	3(011)	2(010)	1(001)
D	D	D	R	D	R	R
1	1	0		1	1	0

r2-----2 ,3, 6, 7

_, 1, 1, 1-----→ must satisfy even parity as per the assumption. As the count of 3,6,7 position is not satisfying even number of 1's therefore r2=1

r4-bit parity



7(111)	6(110)	5(101)	4(100)	3(011)	2(010)	1(001)
D	D	D	R	D	R	R
1	1	0	0	1	1	0

r3-----4 ,5, 6, 7

_, 0, 1, 1-----→ must satisfy even parity as per the assumption. As the count of 5,6,7 position is satisfying even number of 1's therefore r4=0

Therefore, final codeword is ------ 1100110 and sender will send the final code word to receiver.

7	6	5	4	3	2	1
1	1	0	0	1	1	0
			r_3		r_2	r_1

001	1
010	2
011	3
100	4
101	5
110	6
111	7

$r_1 \rightarrow 1, 3, 5, 7$

$r_2 \rightarrow 2, 3, 6, 7$

$r_3 \rightarrow 4, 5, 6, 7$

Error in r_2

Received Code

7	6	5	4	3	2	1
1	1	0	0	1	0	0

$C_1 \rightarrow 0$

$C_2 \rightarrow 1$

$C_4 \rightarrow 0$

Error Bit Posn

$010 = 2$

Error Correction

Error in position 5

$$\begin{array}{lccccccc} & 7 & 6 & 5 & 4 & 3 & 2 & 1 \\ \text{Rx Code} = & 1 & 1 & 1 & 0 & 1 & 1 & 0 \end{array}$$

$C_1 \rightarrow 1, 3, 5, 7$ Even Parity

$= 1$

$C_2 \rightarrow 2, 3, 6, 7$ "

$= 0$

$C_4 \rightarrow 4, 5, 6, 7$ "

$= 1$

$101 = 5$

Error in position 3

$$\begin{array}{lccccccc} & 7 & 6 & 5 & 4 & 3 & 2 & 1 \\ \text{Rx Code} = & 1 & 1 & 0 & 0 & 0 & 1 & 0 \end{array}$$

$C_1 = 1$

$C_2 = 1$

$C_4 = 0$

Error at posn 011

$= 3$

Error Correction

Given data

11110011

Encode using even parity Hamming Code

$k + r + 1 \leq 2^{r}$

$k = 8 \Rightarrow 8 + 4 + 1 \leq 2^{4} = 16.$

$r = 4$

Total bits = 12

$r_1 \rightarrow 1, 3, 5, 7, 9, 11$

$r_2 \rightarrow 2, 3, 6, 7, 10, 11$

$r_4 \rightarrow 4, 5, 6, 7, 12$

$r_8 \rightarrow 8, 9, 10, 11, 12$

r_4	r_3	r_2	r_1	
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12

12	11	10	9	8	7	6	5	4	3	2	1
d	d	d	d	r_8	d	d	d	r_4	d	r_2	r_1
1	1	1	1	0	0	0	1	0	1	1	0

Acknowledge various sources for the images.

Thankyou