Lecture 9
Error Control Coding: Linear Block Code

Error Control Coding.

• Error Control Coding, also known as Channel Coding, is a technique used in Communication Systems to detect and correct errors that may occur during data transmission over unreliable or noisy communication channels.

The purpose of error control coding is to enhance the reliability and robustness of data transmission, ensuring that the received data is as close as possible to the original transmitted data, even in the presence of errors.

There are two types of error control coding techniques:

1. Forward Error correction (FEC)
2. Automatic Repeat Request (ARQ)

Types of ERRor Control Codes

Error Codes

- Block code $(k>1)$
  - Hamming Code
  - R-S Code
  - BCH Code
  - cyclic code
- Convolutional codes $(k=1)$
  - Turbo Code
  - Trellis Code

Block codes $(k>1)$

• In case of block codes from binary sequence k no. of bits are considered for encoding and parity bits are added to get the valid codeword.

• In case of block codes encoding at the current state is independent at the previous stage hence, No MEMORY

Convolutional Codes [k=1]

• In this case 1 bit is considered at a time for encoding and parity bits are added.

• Encoding of the current state depends on the previous state hence, it requires MEMORY

Basics of channel and code word

• Every channel has an upper limit on the rate at which information can be transmitted reliably through the channel

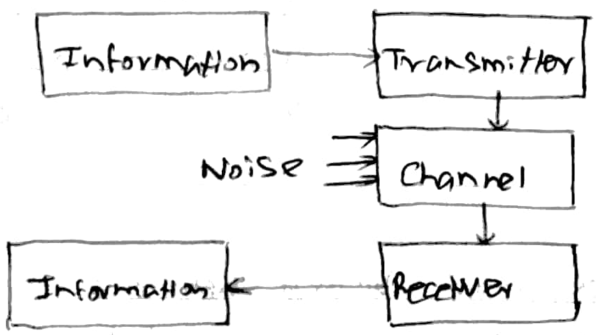
• The limitation of capacity of channel to transmit ~~through~~ information is referred as the channel capacity.

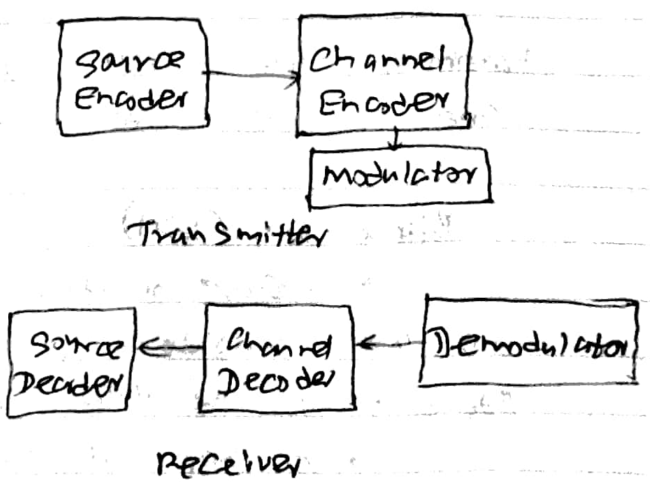
• Due to channel noise there will be error in transmitted bits at receiver side, and to correct error we use block code.

• A block code is a set words that has well defined mathematical property structure and where each word is sequence of fixed no of bits.

• The word belongings to a block code are called as CODE WORDS

Basic structure of communication system





- At channel Encoder we do channel encoding, where we add Block code to solve error.
- At source Encoder we do source decoding, where we reduce redundancy to improve bandwidth utilization.

Basics of Block code

- Information bits = k

$$i = [i_1, i_2, i_3, \ldots\ldots, i_k]$$

- Parity bits / Redundant bits = r

$$P = [P_1, P_2, P_3, \ldots\ldots, P_r]$$

- Here, Total bits of code, $n = k + r$

$$n = [i_1, i_2, i_3, \ldots, i_k, P_1, P_2, P_3, \ldots\ldots P_r]$$

- (n, k) is block code ~~rep~~ representation
- A code word, whose information bits are kept together is systematic
- A code word, whose information bits are not kept together is non-systematic

Important parameters of block code(n, k)

- Total code words required as per n Block codes $= 2^n$
- Total code words required as per k information $= 2^k$
- Total redundant code words required as per r parity bits $= 2^n - 2^k$
- So, Code rate, $R = \frac{k}{n}$
- And n bits block code will be

k information bits   r Redundant bits (Parity bits)

(4, 3) block code $\quad n - k = P$

	3 information Bits (i)			Parity (P)	Code word			
1	0	0	0	0	0	0	0	0
2	0	0	0	1	0	0	0	1
3	0	0	1	0	0	0	1	0
4	0	0	1	1	0	0	1	1
5	0	1	0	0	0	1	0	0
6	0	1	0	1	0	1	0	1
7	0	1	1	0	0	1	1	0
8	0	1	1	1	0	1	1	1
9	1	0	0	0	1	0	0	0
10	1	0	0	1	1	0	0	1
11	1	0	1	0	1	0	1	0
12	1	0	1	1	1	0	1	1
13	1	1	0	0	1	1	0	0
14	1	1	0	1	1	1	0	1
15	1	1	1	0	1	1	1	0
16	1	1	1	1	1	1	1	1

Even Parity check

1, 4, 6, 7, 10, 11, 13, 16

Representation of Block code (n, k)

Information bits $= k$

$$i = [i_1, i_2, i_3, \dots, i_k]$$

Parity bits / Redundant bits $= r$

$$P = [P_1, P_2, P_3, \dots, P_r]$$

Code word, $C = [I, P]$

$$c = [i_1, i_2, i_3, \dots, i_k, P_1, P_2, P_3, \dots P_r]$$

Error code word

$$e = [e_1, e_2, e_3, \dots, e_n]$$

Where, $e_j = 1$ means error and $e_j = 0$ means no error

So valid data = received codeword + error code word

No: ____ Date: __/__/__

Block Code for Parity Check

Definition and Basics for Block Codes for Parity check

- There are a class of error detecting codes that provides the simplest form of error control.
- In this, the codes uses a single parity bit to generate codewords with EVEN or ODD parity.

- In (n,k) block codes, information bits are k.

$$i = [i_1, i_2, i_3, \ldots, i_k]$$

- For EVEN parity bit $P = i_1 \oplus i_2 \oplus i_3 \oplus \ldots \oplus i_k$
- For ODD parity bit $P = i_1 \oplus i_2 \oplus i_3 \oplus \ldots \ldots \oplus i_k \oplus 1$
- Codeword is

$$C = [i_1, i_2, i_3, \ldots, i_k, P]$$

Example of Block code for parity check with Encoding

Example 1: Given the (5,4) even parity block code, find codewords corresponding to I = (1011) and (1010)

- $(5,4) = (n,k)$
- Information bits $k = 4$
- Parity bits $r = n - k = 5 - 4 = 1$
- For Information $I = 1\,0\,1\,1 = I_1\ I_2\ I_3\ I_4$
- For Even Parity

$$P = I_1 \oplus I_2 \oplus I_3 \oplus I_4$$
$$= 1 \oplus 0 \oplus 1 \oplus 1$$
$$= 1$$

So Codeword is = (Information, Parity)

$$C = [i_1, i_2, i_3, i_4, P]$$
$$C = [1, 0, 1, 1, 1]$$

No: Date: / /

For Information $I = 1010 = I_1\ I_2\ I_3\ I_4$

For Even Parity

$$P = I_1 \oplus I_2 \oplus I_3 \oplus I_4$$

$$= 1 \oplus 0 \oplus 1 \oplus 0$$

$$P = 0$$

So codeword is = [Information, parity]

$$C = [i_1, i_2, i_3, i_4, P]$$

$$C = [1, 0, 1, 0, 0]$$

Decoding Stage of Parity check

- At the decoding stage, received codeword is

$$v = [v_1, v_2, v_3, \dots, v_n]$$

- To determine/check whether v is the correct codeword, we do check sum of received codeword.

$$S = v_1 \oplus v_2 \oplus v_3 \oplus \dots \oplus v_n$$

If $S=0$, (Even parity codeword with correct received data)

If $S=1$, (odd parity codeword with correct received data)

Example of Block Code for parity check with Decoding

Example 2: Given the (8,7) even parity block code, Determine whether $v_1 = (1011\ 0110)$ and $v_2 = (0110\ 1001)$ gives parity failures.

$(8,7) = (n,k)$

Information bits $k=7$

Parity bits $r = n-k = 8-7 = 1$

for $v_1 = (10110110)$

Check sum S will be

$$S = D_1 \oplus D_2 \oplus D_3 \oplus D_4 \oplus D_5 \oplus D_6 \oplus D_7 \oplus D_8$$

$$S = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0$$

$$S = 1$$

- For Even received Codeword, it is parity failure
- means there is error in received dat

No: ______ Date: __/__/__

For $v_2 = (01101001)$

Checksum S will be

$$S = D_1 \oplus D_2 \oplus D_3 \oplus D_4 \oplus D_5 \oplus D_6 \oplus D_7 \oplus D_8$$

$$S = 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1$$

$$S = 0$$

- For Even received codeword, it is parity success
- Means there is no error in received data

Block code for Product code

Definition and Basics for Block codes for product

- As we have seen single parity check ~~codes~~ codes have no error correction capability.
- However error correction can be achieved by ~~combining~~ combing two single parity check codes in the form of rectangular array.

Example of Block code for product code

	Data Bits				Row Parity Check	
	1	1	0	1	1	
	0	1	1	0	0	
	1	0	0	0	1	
	0	0	0	0	0	
	1	1	1	0	1	
e.g. parity check just do ...					0	
Column Parity Check	0	1	0	0	1	overall parity

For even parity check, just do mod 2 addition or EXORing operation

For row, this block code is (5,4)

For column, this block code is (7,6)

So product of $(5,4) \times (7,6) = (35, 24)$

~~For row~~

- Here, total 7 rows and 5 Columns. So, total 35 bits
- out of 35 bits 24 bits are information bits
- So given block code is (35,24)
- It is happening as per $(5,4) \times (7,6)$
- It is used to detect and correct one bit error.
- So we cannot identify two error by Product Code, it is Suitable for one error detection & correction

Block code for Repetition code

Definition and Basics for Block Codes for Repetition Code

- These are codes the Codes that repeat information bits two or more times.
- They are block Codes in which the parity bits are set equal to a single information bit and if the no of Parity bits is 'n-1' then the code is referred to as (n,1).

block code is given by $(3,1) = (n,k)$

So $n=3$, $k=1$

So parity bits are

$r = n-k = 3-1 = 2$ bits

- Parity is not used for even/odd.

No : Date: / /

Example of Block Code for Repetition Code

- Let's have example of (3, 1) Repetition code

  $(3,1) = (n, k)$

  Information bits $k=1$

  Parity bits $r = n-k = 3-1 = 2$

- Encoding process

Information bits	Parity Bits	Codeword
0	0 0	0 0 0
1	1 1	1 1 1

- Decoding process
- It is done based on Majority vote Decoding

Received Data	Decoding Decision	Output Data	Infor. i
0 0 0	No error	0 0 0	0
0 0 1	One Bit Error	0 0 0	0
0 1 0		0 0 0	0
1 0 0		0 0 0	0
1 1 1	No error	1 1 1	1
1 1 0	One Bit Error	1 1 1	1
1 0 1		1 1 1	1
0 1 1		1 1 1	1

- Majority of vote for (v_1, v_2, v_3) is taken as per ~~i = v_1 +~~

  $$i = v_1 . v_2 + v_1 . v_3 + v_2 . v_3$$

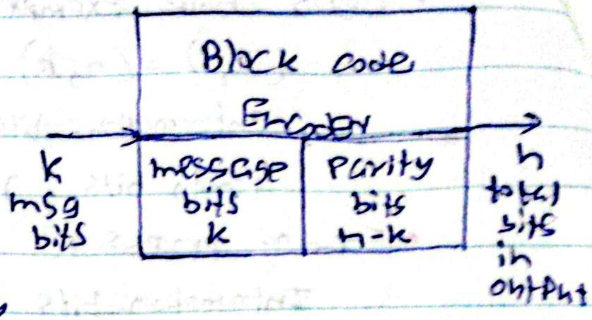
- This is for one bit error detection and one bit error correction

- Here cannot have more than one bit error detection and one bit error correction

No: Date: / /

Terms related to Channel coding

→ Codelength (n)
→ Coderate / Efficiency (η)
→ Code weight
→ Hamming distance
→ minimum Hamming distance (d_{min})
→ Error detection & correction capability



→ Codelength (n) :- Number of bits contained in final codeword is known as codelength

$C_1 = 110101 \Rightarrow n=6$ $C_2 = 1101011 \Rightarrow n=7$

→ Code efficiency (η) :- Ratio of number of msg bits (k) to the number of bits in the codeword (n) is known as code rate / efficiency

$$\eta = \frac{k}{n}$$

→ Code weight :- No of ~~Non~~ non-zero components in the codeword

$C_1 = 110101 \Rightarrow 4$

$C_2 = 1101011 \Rightarrow 5$

Linear Codes basics & property with example

Definition - A Block code is said to be linear code if its codewords satisfy the condition that the sum of any two codewords gives another codeword

i.e. $C_p = C_i + C_k$

Property

i) The all-zero words [0, 0, 0, ... 0] is always a codeword.

ii) Given any ~~two~~ three codewords ~~C_i~~ C_p, C_j and C_k such that

$C_p = C_i + C_k$, then $d(C_i, C_j) = W(P)$

No : Date: / /

(iii) minimum distance of the code.

$$d_{min} = w_{min}$$

– (7,4) Hamming code

$c_1 = 0\ 0\ 0\ 1\ 0\ 1\ 1$ $\boxed{3}$ weight

$c_{10} = 1\ 0\ 1\ 0\ 0\ 1\ 1$ $\boxed{4}$

$c_{11} = 1\ 0\ 1\ 1\ 0\ 0\ 0$ $\boxed{3}$

$$C_{11} = C_1 + C_{10}$$

$C_0 = [0\ 0\ 0\ 0\ 0\ 0\ 0] \rightarrow$ first property satisfies

Second property $\rightarrow$ minimum hamming distance can identify weight of it

$d(C_1, C_{10}) = 3$ | $d(C_1, C_{10}) = 3 = w(C_{11})$

$w(C_{11}) = 3$

Third property $\rightarrow$ minimum distance of the code $d_{min} = w_{min}$

– ~~E~~ $C_{15} = [1\ 1\ 1\ 1\ 1\ 1\ 1]$, $w = 7$

other than C_{15} codes are having weight 3 & 4

$$d_{min} = w_{min}$$

Show that (4,3) even-parity code is a linear and (4,3) odd parity is not linear

$\rightarrow$ (4,3) even parity code

c	d_1	d_2	d_3	P
C_0	0	0	0	0
C_1	0	0	1	1
C_2	0	1	0	1
C_3	0	1	1	0
C_4	1	0	0	1
C_5	1	0	1	0
C_6	1	1	0	0
C_7	1	1	1	1

		w
$\rightarrow$ 0 0 1 1	c_1	2
$\rightarrow$ 0 1 0 1	C_2	2
0 1 1 0	C_3	2

$d_{min}(C_1, C_2) = 2$

$w(C_3) = 2$

$\boxed{d_{min} = w}$

Odd parity (4,3) Code

C	d_1	d_2	d_3	P
C_0	0	0	0	1
C_1	0	0	1	0
C_2	0	1	0	0
C_3	0	1	1	1
C_4	1	0	0	0
C_5	1	0	1	1
C_6	1	1	0	1
C_7	1	1	1	0

$C_1 \rightarrow$ 0 0 1 0 C_1

$C_2 \rightarrow$ 0 1 0 0 C_2

0 1 1 0

- $C_1 + C_2$ is not present in odd parity (4,3) code
- It is not linear block code

Generator matrics to generate codewords in Linear Code

Using a matrix to generate Codewords is a better approach

$$[C] = [i][G]$$

$[C]$ = Codeword

$[i]$ = Information words

$[G]$ = Generator matrix

- The Generator matrix of an (n,k) linear code has 'k' rows and 'n' columns
- Great Generator matrix for $(7,4)$ Code is given by

$$[G] = [I : P]$$

$$= \left[\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\text{Identity matrix } I_k} \; \underbrace{\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}}_{\text{Parity matrix}} \right]$$

No: Date: / /

$\rightarrow [C] = [i][G]$

Example - Generate codeword for $i = (1\ 1\ 1\ 0)$ with $(7,4)$ generator matrix code.

$$G = \left[\begin{array}{cccc:ccc} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right]$$

$\Rightarrow$ message $[i] = [1\ 1\ 1\ 0]$

$$C = [i][G]$$

$$= [1\ 1\ 1\ 0]\left[\begin{array}{cccc:ccc} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right]$$

$$= [1\ 1\ 1\ 0\ 1\ 0\ 0]$$

Example - Determine the set of codewords for the $(6,3)$ code with generator matrix

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

$\rightarrow n \rightarrow 6$

$k \rightarrow 3$

$\rightarrow$ message bits $= 3$

m_0	m_1	m_2
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

$\rightarrow C = [i][G]$

$$\rightarrow C_0 = [0\ 0\ 0]\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

$$C_0 = [0\ 0\ 0\ 0\ 0\ 0]$$

$$C_1 = [0\ 0\ 1]\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

$$= [0\ 0\ 1\ 1\ 1\ 0]$$

No: Date:

	m_0	m_1	m_2	P_0	P_1	P_2
C_0	0	0	0	0	0	0
C_1	0	0	1	1	1	0
C_2	0	1	0	1	0	1
C_3	0	1	1	0	1	1
C_4	1	0	0	0	1	1
C_5	1	0	1	1	0	1
C_6	1	1	0	1	1	0
C_7	1	1	1	0	0	0

$$[C] = [m : P_c]$$
$$[P_c] = [m][P]$$

$$[G] = [I : P]$$

Systematic Generator matrix in Linear Block Codes

- A Generator matrix $[G] = [I_k : P]$ is said to be in a Systematic form if it generates the systematic codewords.

- Hence $\qquad [C] = [i][G]$
$\qquad = (m, P)$

$[I_k] \rightarrow k \times k$ matrix
$[P] \rightarrow k \times (n-k)$ matrix
$[G] \rightarrow k \times n$ matrix

- In these matrix Information bits are placed together Codeword

$$[C] = [i][G]$$
$$= [i_1, i_2, i_3, i_4] \begin{array}{c} \begin{array}{cccccccc} i_1 & i_2 & i_3 & i_4 & P_1 & P_2 & P_3 \end{array} \\ \left[\begin{array}{cccc:ccc} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right] \end{array}$$

- So when you get codeword
$$C = (i_1, i_2, i_3, i_4, P_1, P_2, P_3)$$

No :   Date: / /

- Identity matrix keeps information together
- Parity matrix generates parity bits

$$P_1 = i_1 + i_2 + i_3$$

$$P_2 = i_2 + i_3 + i_4$$

$$P_3 = i_4 + i_2 + i_4$$

$$G = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} = [I_k : P]$$

- Determine Systematic form of G
- Generate Codeword for information [0 1 1] with systematic G & Non-Systematic G

$$[G] = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} \xrightarrow{\text{Add } R_2 \text{ to } R_3} \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & \phi \end{bmatrix}$$

For being Systematic first row should be 1 0 0 for identity. second row, third row 0 0 1

Add R_3 to R_1

$$\begin{bmatrix} 1 & 0 & 0 & \phi & \phi \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & \phi \end{bmatrix}$$

I   P

$$G = [I_k : P]$$

→ For Non-Systemetic [G]

$$C = [i][G]$$

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

$$= [0\ 0\ 1\ 1\ 1]$$

→ For Systematic (G)

$$C = [i][G]$$

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

$$= [\underbrace{0\ 1\ 1}_{\text{Information}}\ \underbrace{1\ 0}_{\text{Parity}}]$$

No: Date: / /

Parity check matrices in Linear block codes with Example

- From Generator matrix G $[G] = [I_k | P]$ we can identify parity matrix
  - By taking P^T we can make parity check matrix $[H]$

$$[H] = [P^T : I_{n-k}]$$

- Parity check matrix is used at Rx to decode data

Example- Generate Parity check matrix for (7,4) code

$$[G] = \left[\underbrace{\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}}_{\text{Identity matrix } I} \;\middle|\; \underbrace{\begin{matrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{matrix}}_{\text{Parity matrix } P}\right]$$

$$\rightarrow P = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \Rightarrow P^T = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix}$$

$$\rightarrow I_{n-k} = I_{7-4} = I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\rightarrow [H] = [P^T : I_{n-k}]$$

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

Proove that GH^T and $CH^T = 0$

$$\Rightarrow GH^T = [I_k | P] \left[P^T | I_{n-k} \right]^T$$

$$= [I_k | P] \left[\frac{P}{I_{n-k}} \right]$$

$$= \left[I_k P + P I_{n-k} \right]$$

$$= P + P$$

$P \to 0, 1$ For mod-2 add, $0+0=0$, $1+1=0$

$$\Rightarrow \boxed{GH^T = 0}$$

$$\Rightarrow CH^T = [i][G][H^T]$$

$$= [i]\, 0$$

$$= 0$$

$$\boxed{CH^T = 0}$$

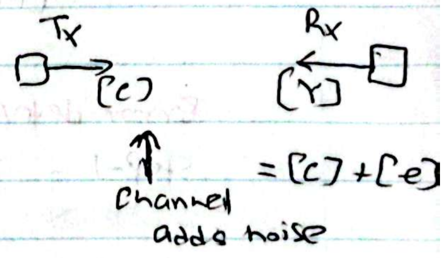
Error Syndromes in Linear block codes with Example

- If received codeword is $[Y]$
- then error syndromes

$$[S] = [Y][H^T]$$

- Hence, received codeword $[Y]$

$$[Y] = [c] + [e]$$



Here where, $[c]$ = codeword

$[e]$ = error

- If error $[e] = 0$, $[Y] = [c]$

So,

$$[S] = [c][H^T] = 0$$

No: Date:

Find the error syndromes of $v_1 = (1101101)$, for $H^T = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, where v_1 is received codeword also calculate the errored bit

→ Received codeword

$$v_1 = [1\ 1\ 0\ 1\ 1\ 0\ 1]$$

→ Error Syndrome

$$[S] = [v_1][H^T]$$

$$= \begin{array}{c} [1\ 1\ 0\ 1\ 1\ 0\ 1] \\ \scriptstyle 1\ 2\ 3\ 4\ 5 \end{array} \begin{array}{c} \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ \boxed{1 \quad 0 \quad 0} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{array} \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array}$$

$$= [1\ 0\ 0]$$

$Y = [1101101]$

$e = [0000100]$

$X = Y + e$

$= [1101001]$

Error detection and correction capability of Linear Block Code

Step-1 - Identify d_{min} [minimum frequency Hamming distance]

Error detection capacity of Linear block code

$\Rightarrow d_{min} \geq s + 1$

→ where, s = error detection capacity

Error correction capacity of linear block code

$\Rightarrow d_{min} \geq 2t + 1$

→ where, t = error correction capacity

No : Date: / /

- If minimum hamming distance of Linear block code is 3. Find ~~LTE~~ LBC code detection & correction capability

→ $d_{min} = 3$

→ ~~error~~ for error detection

$\Rightarrow d_{min} \geq s + 1$

$\Rightarrow 3 \geq s + 1$

$\Rightarrow 2 \geq s$

$\Rightarrow s \leq 2$

→ this code can detect 2 bit error

→ For error correction

$d_{min} \geq 2t + 1$

$\Rightarrow$ ~~d_{min}~~ $3 \geq 2t + 1$

$\Rightarrow 2 \geq 2t$

$\Rightarrow 1 \geq t$

$\Rightarrow t \leq 1$

→ this code can correct 1 bit error

Linear Block Codes complete example

- For a (6, 3) code, the generator matrix G is

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

Find (a) All corresponding code vectors

(b) minimum Hamming distance ~~dm~~ d_{min}

(c) Error detection & Error correction Capability

(d) Parity check matrix

(e) find error if received code is [1 0 0 0 1 1]

Atlas

CamScanner

$\rightarrow \quad G = [I_k : P]$

$$I_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad P = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

$\Rightarrow [C] = [i][G]$

$\quad = [m : P_c]$

$\Rightarrow [P_c] = [i_m][P]$

$\cancel{[P_c]} \rightarrow [P_0 \;\; P_1 \;\; P_2] = [i_0 \;\; i_1 \;\; i_2] \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$

$P_0 = (i_0 \oplus i_2)$

$P_1 = (i_1 \oplus i_2)$

$P_2 = (i_0 \oplus i_1)$

C	i_0	i_1	i_2	P_0	P_1	P_2	w
C_1	0	0	0	0	0	0	–
C_2	0	0	1	1	1	0	3
C_3	0	1	0	0	1	1	3
C_4	0	1	1	1	0	1	4
C_5	1	0	0	1	0	1	3
C_6	1	0	1	0	1	1	4
C_7	1	1	0	0	1	0	4
C_8	1	1	1	0	0	0	3

(b) To minimum hamming distance, we need to see weight of each codeword

So. $d_{min} = 3$

No: Date:

(c) Error detection and Error correction Capability, we can have based on minimum hamming distance

$\Rightarrow d_{min} = 3$

$\Rightarrow$ Error detection

$\Rightarrow d_{min} \geq s+1$

$\Rightarrow 3 \geq s+1$

$\Rightarrow s \leq 2$

$\Rightarrow$ It can detect 2 bit error

$\Rightarrow$ Error correction

$\Rightarrow d_{min} \geq 2t+1$

$\Rightarrow 3 \geq 2t+1$

$\Rightarrow t \leq 1$

$\Rightarrow$ It can correct 1 bit error

(d) $H = [P^T : I_{n-k}]$

$\Rightarrow P^T = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$ $\quad I_{6-3} = I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

$\Rightarrow H = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$

(e) $\Rightarrow [S] = [Y][H^T]$

$[Y] = [1\ 0\ 0\ 0\ 1\ 1]$

$[H^T] = \begin{array}{c} 1 \\ 2 \\ 3 \\ \\ \\ \\ \end{array} \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

$S = YH^T$

$= [1\ 0\ 0\ 0\ 1\ 1] \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 3rd

position of syndrome

$S = [1\ 1\ 0]$

~~$\Rightarrow$ Here~~

Error is happening at 3rd bit

$\Rightarrow$ Here error happened at 3rd bit

$\cancel{e = [0\ 0\ 1]}$ $\quad e = [0\ 0\ 1\ 0\ 0\ 0]$

$Y = [1\ 0\ 0\ 0\ 1\ 1]$

To extract original information

$X = e + Y$

$= [1\ 0\ 1\ 0\ 1\ 1] \Rightarrow$ actual transmitted signal