Chapter 5 Part 3

Channel Coding

CSE4214 Digital Communications

Error-Detecting and Correcting Capability

Weight and Distance of Binary Vectors

- Hamming weight (w)
  - Number of non-zero elements in a cordword
- Hamming distance (d)
  - Number of elements in 2 codewords in which they differ

  Example :

  U = 1 0 0 1 0 1 1 0 1 → w(U)=5

  V = 0 1 1 1 1 0 1 0 0 → w(V)=5, d(U,V)=6

  U + V = 1 1 1 0 1 1 0 0 1 → w(U+V)=6
- Hamming weight of a codeword is equal to its Hamming distance from the all-zeros vector

3

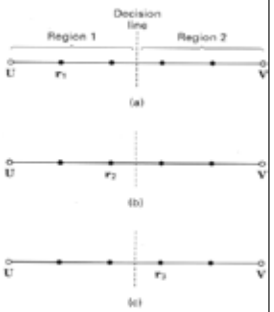
Minimum Distance of a Linear Code

- The smallest distance among all pairs of codeword (d_{min})
- Determine the minimum distance
  - Examine the weight of each codewords, and pick the minimum, that is d_{min}
- Minimum distance gives a measure of the code's minimum capability and characterizes the code's strength.

4

Error Detection and Correction

- An example
  - Assume d_{min} between U and V = 5
  - Case r1, 1 bit error from U, decoder will correct the vector r1 to U code words
  - Case r2, 2 distances error from U and 3 distance errors from V, decoder will choose U
  - Case r3, 3 distances error from U and 2 distances error from V, decoder will choose V



Error Detection and Correction (2)

- The decoder corrects the vector to the nearest code word
- The error-correcting capability t of a code is defined as:

$$t = \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor$$

where $\lfloor x \rfloor$ means the largest integer not to exceed x.

- The error-detecting capability can be defined by :

$$e = d_{min} - 1$$

Simultaneous Error Correction & Detection

- A code can be used for the simultaneous correction of α errors and detection of β errors, where $\beta \geq \alpha$, provide that its minimum distance is:

  $d_{min} \geq \alpha+\beta+1$
  - When t or fewer errors occur, the code is capable of detecting and correcting them
  - When more than t but fewer than e+1 errors occur the code is capable of detection them but correcting only a subset of them.

Example

A code with d_{min}=7 (t=3, e=6) can be used to simultaneously detect and correct in any one of the following ways:

Detect (β)	Correct (α)
3	3
4	2
5	1
6	0

Erasure Correction

- Some receiver might be designed to declare a symbol erased when it is received ambiguously.
  - Given minimum distance d_{min}, any pattern of p or fewer erasures can be corrected if $d_{min} \geq p+1$.
  - Any pattern of α errors and γ erasures can be corrected simultaneously if $d_{min} \geq 2\alpha+\gamma+1$

Activity 1

Consider the codeword set of (6,3), suppose the codeword 110011 was transmitted and that two leftmost digits were declared by the receiver to be erasures. Verify that the received flawed sequence xx0011 can be corrected.

Message vector	Codeword
000	000000
100	110100
010	011010
110	101110
001	101001
101	011101
011	110011
111	000111

Usefulness of the Standard Array

The Standard Array

- Standard array for $[n,k]$ code is a 2^{n-k} by 2^k matrix
  - The 1st row list all codewords with 0 codewords on the extreme left
  - Each row is a coset with the coset leader in the first column
  - The entry in the i-th row and j-th column is the sum of the i-th coset leader and j-th codeword

An Example of (6,3) Code

000000	110100	011010	101110	101001	011101	110011	000111
000001	110101	011011	101111	101000	011100	110010	000110
000010	110110	011000	101100	101011	011111	110001	000101
000100	110000	011110	101010	101101	011001	110111	000011
001000	111100	010010	100110	100001	010101	111011	001111
010000	100100	001010	111110	111001	001101	100011	010111
100000	010100	111010	001110	001001	111101	010011	100111
010001	100101	001011	111111	111000	001100	100010	010110

Estimating Code Capability

- Standards array allow the visualization of important performance issues, such as possible trade-offs between error correction and detection
- ***Hamming bound*** is one of the bounds on error-correction capability

Number of parity bits:

$$n-k \geq \log_2\left[1+\binom{n}{1}+\binom{n}{2}+\ldots+\binom{n}{t}\right]$$

or number of cosets:

$$2^{n-k} \geq \left[1+\binom{n}{1}+\binom{n}{2}+\ldots+\binom{n}{t}\right]$$

Estimating Code Capability

- ***Plotkin bound*** is an upper bound on the *t*-bit error-correction capability

$$d_{\min} \leq \frac{n \times 2^{k-1}}{2^k - 1}$$

- In general, a linear (n,k) code must meet all upper bounds involving error correction capability (or minimum distance).
  - For high-rate code, if the Hamming bound is met, then Plotkin bound will also be met.
  - For low-rate code, it is other way around.

Example: No. of Cosets Required

- (127,106) code
  - 2^{127} n-tuples in the space
  - Topmost row: 2^{106} codeword (columns)
  - Leftmost column: $2^{n-k}=2^{21}=2{,}097{,}152$ coset leaders (rows)
    - 2,097,152 cosets = 2,097,152 error patterns, which can be corrected

TABLE 6.3 Error-Correction Bound for the (127, 106) Code

Number of Bit Errors	Number of Cosets Required	Cumulative Number of Cosets Required
0	1	1
1	127	128
2	8,001	8,129
3	333,375	341,504
4	10,334,625	10,676,129

- Hence, this Humming bound can guarantee the correction up to 3 bits only.

Design of a (*n*,*k*) Code

- How to choose *n* and *k*?
  - Assume required error-correcting capability is at least t=2, then $d_{min}=2t+1=5$
  - Assume k=2, i.e. $2^k=4$ codewords
  - If Hamming bound is used, the min n=7.
  - Checking for Plotkin bound, $n \geq 7.5$, so n=8
  - The minimum dimensions of the code are (8,2).

17

Designing of (8,2) Code

- How to determine codeword?
  - The number of code is $2^k=4$, and each code is 8-bit
  - The all-zero vector must be one of the codeword
  - The closure property must be met
  - Since d_{min}=5, the weight of each codeword, except for all-zero code), must also be at least 5.
  - Assume the code is systematic, the rightmost 2 bits of each codeword are the message bits

Message	Codewords
00	00000000
01	11110001
10	00111110
11	11001111

18

CSE4214 Digital Communications

Cyclic Codes

Definition

- Definition
  - A code is cyclic if it is a linear code
  - Any cyclic shift of a codeword is also a codeword, i.e. $U=(u_0,\dots u_{n-1})$ in subspace S, $U^{(1)}=[u_{n-1}\ u_0 \dots u_{n-2}]$ is also in S.
- Example
  - Code U= {0000, 1111} is cyclic
  - Code U = {000, 101, 011, 110} is cyclic.
  - The binary linear code {000, 100, 011, 111} is not a cyclic.

20

Algebraic Structure

- Express codewords in polynomial form.

$$U(X) = u_0 + u_1 X + u_2 X^2 + \ldots + u_{n-1} X^{n-1}$$

  - If U(X) is an (*n*-1) degree codeword polynomial, then $U^{(i)}(X)$, the remainder resulting from dividing $X^i U(X)$ by X^n+1, is also a codeword.
  - Simply, $X^i U(X) = q(X)(X^n+1) + \underbrace{U^{(i)}(X)}_{remainder}$
  - In terms of modulo expression

$$U^{(i)}(X) = X^i U(X) \text{ modulo } (X^n+1)$$

Activity 2

Let U = 1 1 0 1, for *n*=4. Express the codeword in polynomial form, and solve for the third end-around shift of the codeword.

Cyclic Code Properties

- Generate a cyclic code using a generator polynomial
  - The generator polynomial g(X) for an (*n*, *k*) cyclic code is unique and is of the form

$$g(X) = g_0 + g_1 X + g_2 X^2 + \ldots + g_p X^p$$

  - The message polynomial m(X) is written as

$$m(X) = m_0 + m_1 X + m_2 X^2 + \ldots + m_{n-p-1} X^{n-p-1}$$

  - Every codeword polynomial in the (*n*,*k*) cyclic code can be expressed as

$$U(X) = (m_0 + m_1 X + m_2 X^2 + \ldots + m_{n-p-1} X^{n-p-1}) g(X)$$

- The generator polynomial g(X) of an (*n*, *k*) cyclic code is a factor of $X^n + 1$, i.e. $X^n + 1 = g(X)\,h(X)$.

Cyclic Code Example

- $X^7 + 1 = (1 + X + X^3)(1 + X + X^2 + X^4)$
  - Let $g(X) = 1 + X + X^3$ as a generator polynomial, $n - k = 3$, we can generate an (n, k) = (7, 4) cyclic code.
  - Let $g(X) = 1 + X + X^2 + X^4$ as a generator polynomial, $n – k = 4$, we can generate an (*n*, *k*) = (7, 3) cyclic code.
- If g(X) is a polynomial of degree *n* - *k* and is factor of $X^n + 1$, then g(X) uniquely generates an (*n*, *k*) cyclic code.

Error Detection

- Assume U(X) is transmitted and Z(X) is received

  U(X) = m(X) g(X)

  Z(X) = U(X) + e(X)

  where e(X) is the error pattern polynomial
- The decoder tests whether Z(X) is a codeword polynomial, i.e. whether it is divisible by g(X) with a zero remainder
  - Z(X)=q(X)g(x)+S(X), syndrome S(X) is the remainder of Z(X) divided by g(X)
  - Also U(X) + e(X)=q(X)g(x)+S(X)

  → e(X)=[m(X)+q(X)]g(X)+S(X)

  Syndrome is the remainder of e(X) divided by g(X)

25

Error Detection

- S(X) = Z(X) modulo g(X)
- S(X) = e(X) module g(X)
- The syndrome contains the information needed for the correction of the error pattern.
- The syndrome calculation is accomplished by a division circuit → feedback shift register

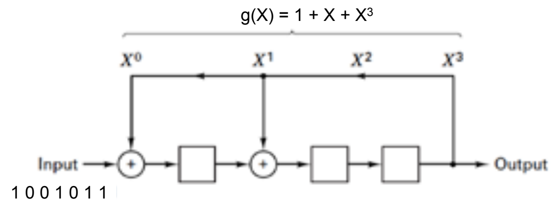
26

Activity 3

Let the received signal is Z = 1 0 0 1 0 1 1. Assume that the generator is g= 1 1 0 1. Calculate the syndrome.

27

Implementation with Shift Registers



Input queue	Shift number	Register contents
1001011	0	000
100101	1	100
10010	2	110
1001	3	011
100	4	011
10	5	111
1	6	101
–	7	$\boxed{000}$ Syndrome

28

CSE4214 Digital Communications

Well Known Block Codes

Hamming Codes

- Invented by Richard Hamming in 1950
- Simple class of block codes characterized by the structure
  $(n, k) = (2^m - 1, 2^m - 1 - m)$ where m = 2, 3, ….
  - Have a minimum distance of 3.
  - Capable of correcting all single errors or detecting all combinations of two or fewer errors within a block
  - The bit error probability can be written as

$$P_B \approx \frac{1}{n}\sum_{j=2}^{n} j \binom{n}{j} p^j (1-p)^{n-j}$$

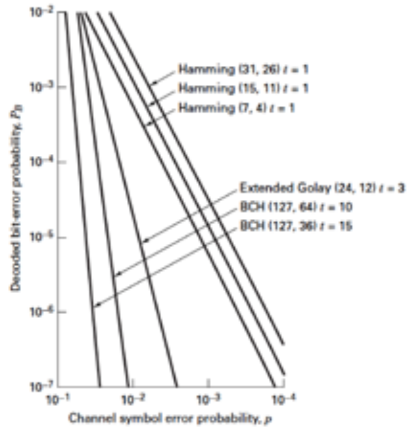
or the following equivalent equation

$$P_B \approx p - p(1-p)^{n-1}$$

Extended Golay Code

- (24, 12) extended Golay Code, formed by adding an overall parity bit to the (23, 12) code.
- The added parity bit increases the minimum distance d_{min} from 7 to 8.
- These codes are considerably more powerful than the Hamming codes.
- The error performance of the extended Golay code is seen to be significantly better than that of the Hamming codes.

Hamming Codes

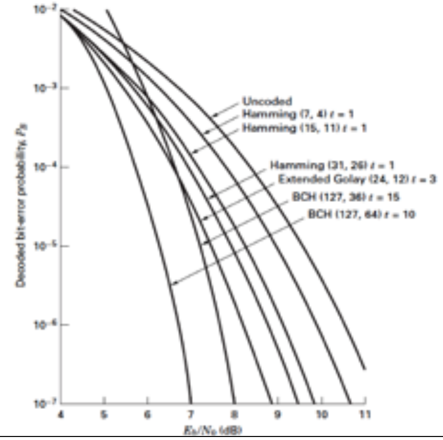


BHC Codes

- Boss-Chadhuri-Hocqenghem (BCH) codes are generalization of Hamming codes that allow multiple error correction.
- They are a powerful class of cyclic codes that provides a large selection of block lengths, code rates, alphabet sizes, and error-correcting capability.

33

BCH Codes



34