Stanford University

ENGR 76

Information Science and Engineering

Lecture 14: Hamming Bound and Hamming Codes

Siddharth Chandak

Recap

Error Correction Codes

- Adding redundancy (extra bits) so that errors caused by the channel can be handled at the receiver

Error Detection and Correction

Detection:

- Receiver checks consistency
  - Could this have been transmitted?
- If not valid: error declared

Correction:

- Receiver finds the most likely transmitted codeword
  - What was actually transmitted?
- Can correct errors up to a certain limit

Repetition Codes

Map 0 to 000 and 1 to 111

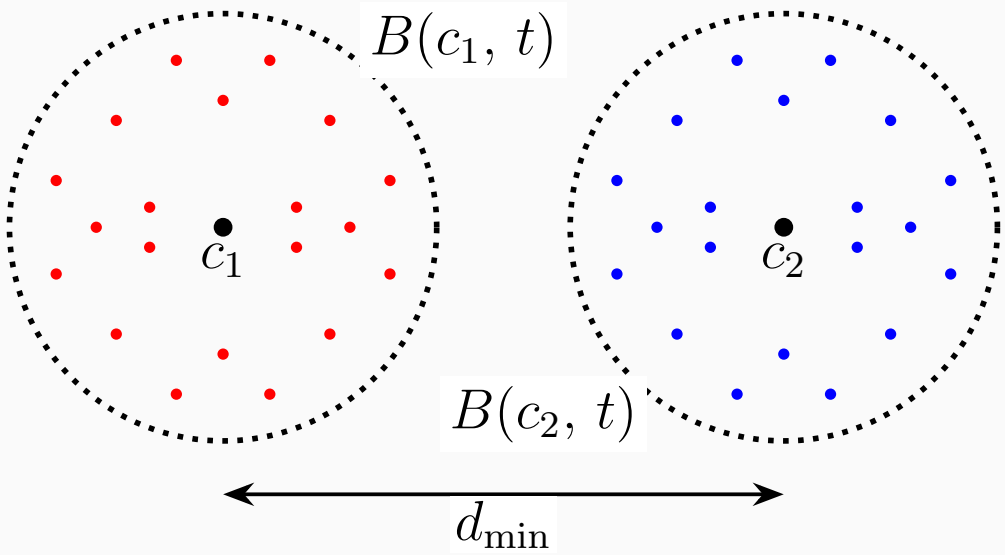
- Information Sequence: 1011
- Encoded Sequence: 111000111111
- Received Sequence: 101100111100
- Decoded Sequence: 111000111000
- Decoded Information: 1010

Minimum Distance of the Code

- The minimum Hamming distance between any two codewords
- Denoted by d_{min}
- Example:
  - Suppose $\mathcal{C} = \{00000, 00111, 11100, 11011\}$
    - $d_H(00000, 00111) = 3$
    - $d_H(00000, 11100) = 3$
    - $d_H(00000, 11011) = 4$
    - $d_H(00111, 11100) = 4$
    - $d_H(00111, 11011) = 3$
    - $d_H(11100, 11011) = 3$
  - $d_{min} = 3$

Error Correction

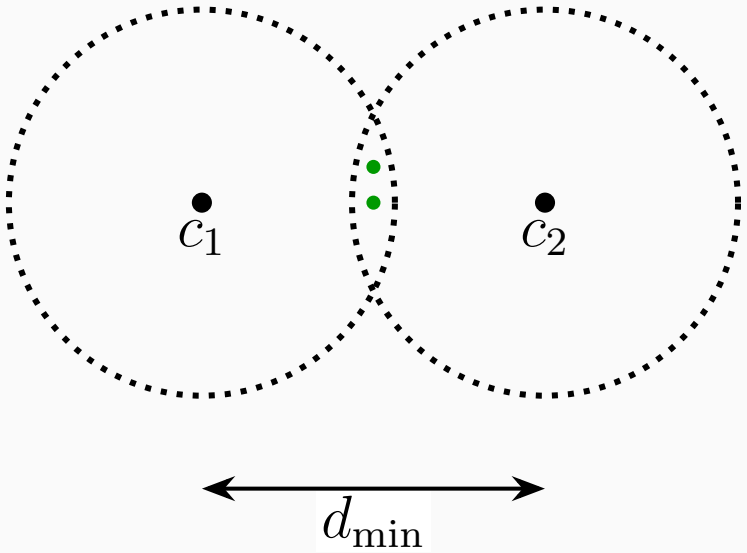
- How many bit flips can a code with minimum distance d_{min} correct?



- Red points: bit strings within Hamming distance t of c_1
- Blue points: bit strings within Hamming distance t of c_2
- We can correct up to t bit flips as long as these balls do not intersect

Error Correction

- How many bit flips can a code with minimum distance d_{min} correct?



- Cannot correct errors in case of intersection
- Non-intersection requires $d_{min} > 2t$

Error Correction

Theorem (Error Correction)

A code can correct up to t bit flips if and only if

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

Equivalently, a code with minimum distance $d_{\min}$ can correct up to

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

bit flips.

- Here $\lfloor x \rfloor$ denotes the floor operator, largest integer smaller than or equal to x
- $\lfloor 1 \rfloor = 1$ and $\lfloor 1.5 \rfloor = 1$

Figures of Merit

- n: Number of bits in each codeword
- k: Number of information bits
  - $M = 2^k$ is the number of messages or codewords - size of the code
- d_{min}: Minimum distance of the code
- Rate $= \frac{k}{n}$
  - Ratio of information bits to the total number of bits in a codeword

Examples

- $00 \to 00000, 01 \to 00111, 10 \to 11100, 11 \to 11011$
- $n = 5$ (number of bits in each codeword)
- $k = 2$ (two information bits)
- $M = 4$ (4 codewords)
- Rate $= 2/5$
- $d_{min} = 3$ (distance between codewords)

Design Goals

We would like:

- Larger k and M: more information bits
- Small n: Want smaller transmissions
- Larger rate
- Larger d_{min}: Better error correction capabilities

Hamming Code

Hamming Code

- A method to generate codes with $d_{min} = 3$
- **Hamming Codes**
  - Developed in 1940s by Richard Hamming
  - First "practical" error correction code
  - Still in use where fast and low-complexity decoding is required and error rate is low
  - Memory hardware

(7,4) Hamming Code

- $n = 7$: Codewords are of length 7
- $k = 4$: 4 information bits
  - $b_1 b_2 b_3 b_4$ are the information bits
- Need to understand the XOR $\oplus$ operation first

XOR operation

- $\oplus$ operation
- Output is 1 if bits are different, 0 if bits are same
  - $0 \oplus 0 = 0$
  - $0 \oplus 1 = 1$
  - $1 \oplus 0 = 1$
  - $1 \oplus 1 = 0$
- Intuitive way to think:
  - Output is 1 if number of 1's in input bits is odd
  - Output is 0 if number of 1's in input bits is even
- Examples:
  - $0 \oplus 0 \oplus 1 = ?$
  - $1 \oplus 1 \oplus 1 = ?$
  - $1 \oplus 0 \oplus 1 = ?$
  - $0 \oplus 0 \oplus 0 = ?$

XOR operation

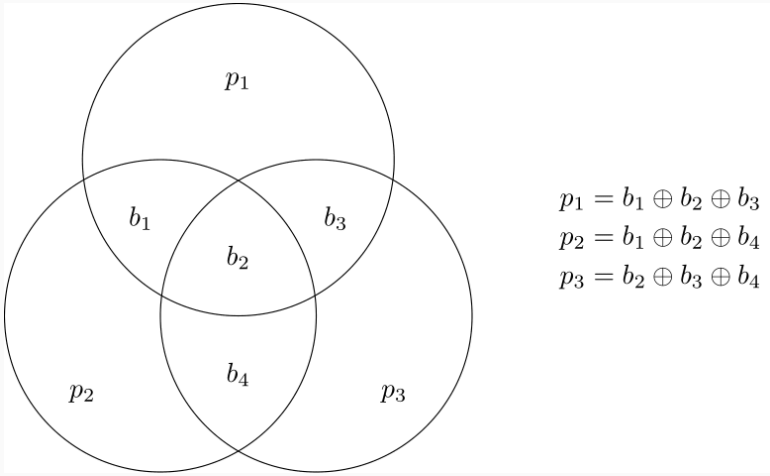
- $\oplus$ operation
- Output is 1 if bits are different, 0 if bits are same
  - $0 \oplus 0 = 0$
  - $0 \oplus 1 = 1$
  - $1 \oplus 0 = 1$
  - $1 \oplus 1 = 0$
- Intuitive way to think:
  - Output is 1 if number of 1's in input bits is odd
  - Output is 0 if number of 1's in input bits is even
- Examples:
  - $0 \oplus 0 \oplus 1 = 1$
  - $1 \oplus 1 \oplus 1 = 1$
  - $1 \oplus 0 \oplus 1 = 0$
  - $0 \oplus 0 \oplus 0 = 0$

(7,4) Hamming Code - Encoding

- 4 information bits: $b_1 b_2 b_3 b_4$
- Corresponding codeword: $b_1 b_2 b_3 b_4 p_1 p_2 p_3$ (7 bits)
  - p_1, p_2, p_3 are three parity bits
  - $p_1 = b_1 \oplus b_2 \oplus b_3$
  - $p_2 = b_1 \oplus b_2 \oplus b_4$
  - $p_3 = b_2 \oplus b_3 \oplus b_4$
- Can be verified that $d_{min} = 3$

(7,4) Hamming Code - Encoding

- 4 information bits: $b_1 b_2 b_3 b_4$
- Corresponding codeword: $b_1 b_2 b_3 b_4 p_1 p_2 p_3$ (7 bits)



- Each circle will have an even number of 1's. Example:
  - If b_1, b_2, b_3 have odd number of 1's, then p_1 is 1
  - If b_1, b_2, b_3 have even number of 1's, then p_1 is 0

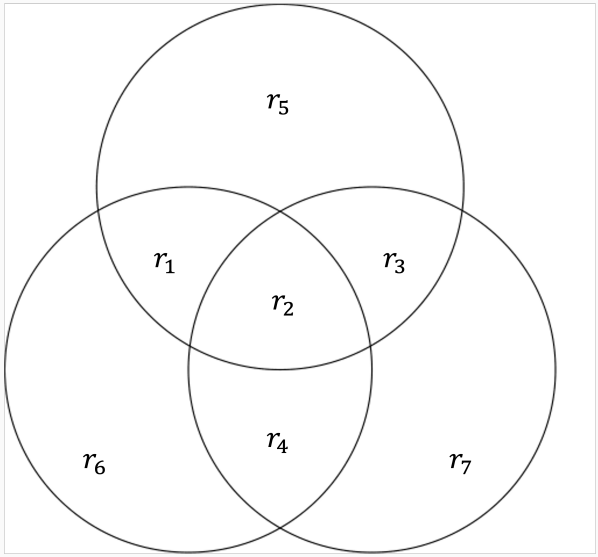
(7,4) Hamming Code - Decoding

- Decoding can be done using the standard minimum distance decoding method as well
- But there exists a more efficient way as well
  - Parity-check or syndrome decoding
- Suppose received bits are $r_1 r_2 r_3 r_4 r_5 r_6 r_7$

Transmitted:	b_1	b_2	b_3	b_4	p_1	p_2	p_3
Received:	r_1	r_2	r_3	r_4	r_5	r_6	r_7

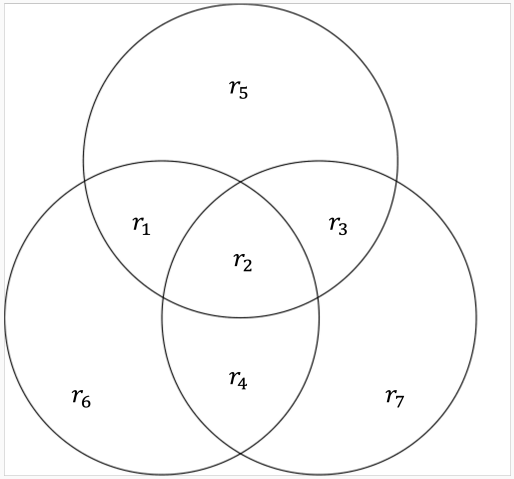
(7,4) Hamming Code - Syndrome Decoding

- For codewords, each circle has an even number of 1's
- Decoded codeword
  - Bit sequence with minimum distance from $r_1r_2r_3r_4r_5r_6r_7$ such that all circles have even number of 1's



(7,4) Hamming Code - Syndrome Decoding

- If all circles have an even number of 1's
  - the received bits are the codeword
- If only one circle has an odd number of 1's
  - error in the bit which affects only that circle
- If only two circles have an odd number of 1's
  - error in the bit which affects only those two circles
- If all three circles have an odd number of 1's
  - error in the bit which affects all three circles



(7,4) Hamming Code - Syndrome Decoding

Examples: Suppose received bits are:

- 0101100
- 1110110
- 1011101
- 0111110

(7,4) Hamming Code - Syndrome Decoding

0101100

- All circles have even number of 1's
- No bit flips
- Decoded codeword: 0101100
- Decoded information bits: 0101

1110110

- Left circle has odd number of 1's
- Bit flip in sixth bit
- Decoded codeword: 1110100
- Decoded information bits: 1110

1011101

- Top and right circle have odd number of 1's
- Bit flip in third bit
- Decoded codeword: 1001101
- Decoded information bits: 1001

0111110

- All circles have odd number of 1's
- Bit flip in second bit
- Decoded codeword: 0011110
- Decoded information bits: 0011

(7,4) Hamming Code - Syndrome Decoding

- Suppose received bits are $r_1 r_2 r_3 r_4 r_5 r_6 r_7$
- Syndrome decoding explained formally
- Then compute syndrome bits
  - $s_1 = r_1 \oplus r_2 \oplus r_3 \oplus r_5$
  - $s_2 = r_1 \oplus r_2 \oplus r_4 \oplus r_6$
  - $s_3 = r_2 \oplus r_3 \oplus r_4 \oplus r_7$
- Decoding based on value of $s_1 s_2 s_3$

(7,4) Hamming Code - Decoding

- $d_{min} = 3 \implies$ up to 1 bit flip can be corrected

$s_1 s_2 s_3$	bit flipped	decoded output $\hat{b}_1 \hat{b}_2 \hat{b}_3 \hat{b}_4$
000	none	$r_1 r_2 r_3 r_4$
110	b_1 (position 1)	$\overline{r}_1 r_2 r_3 r_4$
111	b_2 (position 2)	$r_1 \overline{r}_2 r_3 r_4$
101	b_3 (position 3)	$r_1 r_2 \overline{r}_3 r_4$
011	b_4 (position 4)	$r_1 r_2 r_3 \overline{r}_4$
100	p_1 (position 5)	$r_1 r_2 r_3 r_4$
010	p_2 (position 6)	$r_1 r_2 r_3 r_4$
001	p_3 (position 7)	$r_1 r_2 r_3 r_4$

- $\overline{r}$ denotes bit complement, i.e., $r = 0 \implies \overline{r} = 1$ and $r = 1 \implies \overline{r} = 0$

How Good is the (7,4) Hamming code?

- Length of each codeword is $n = 7$ bits
- $d_{min} = 3$
- Number of codewords (or size of code) is $M = 2^k = 2^4 = 16$
- Want as large M as possible
- How many codewords can we have if we fix $n = 7$ and $d_{min} = 3$?

Fundamental Bound

- Each codeword is a binary string of length n
- We are choosing codewords such that each of them are at least 3 bit flips apart
- Can expect a fundamental upper bound
  - **Hamming Bound**

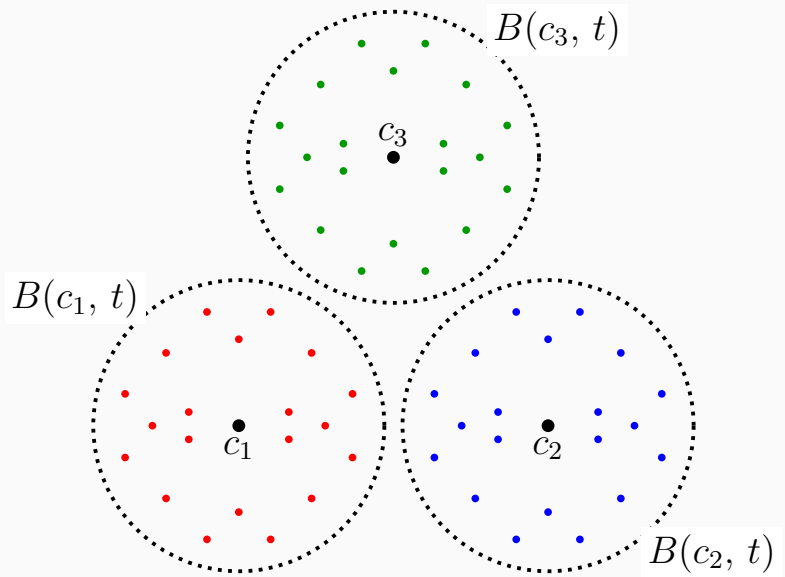
Hamming Bound

Hamming Bound

- Consider length of codeword n bits and $d_{min} = 3$
- What is the maximum number of codewords M possible?
- **Sphere Packing Argument**

Sphere Packing Argument

- Recall that if $d_{min} = 3$, then we can correct $t = \lfloor \frac{d_{min}-1}{2} \rfloor = 1$ errors
- We need non-overlapping Hamming balls of radius 1 around each codeword



Sphere Packing Argument

- Let number of strings in Hamming ball $B(c, 1)$ be given by $|B(c, 1)|$
- Each codeword has one such ball
- Total number of strings of length n is 2^n
- We need

$$M|B(c, 1)| \leq 2^n$$

- $|B(c, 1)| = ?$

Number of Strings in Hamming Balls

- Number of strings in $B(c, 1)$
- All bit strings which are up to one bit flip away from codeword c
  - Zero bit flips away: original codeword: 1 string
  - One bit flip away: n strings (one out of n bits changes)
- $|B(c, 1)| = n + 1$
- Number of codewords:

$$M \leq \frac{2^n}{n+1}$$

Hamming Bound for $d_{min} = 3$

Theorem

For any code of length n bits with minimum distance $d_{min} = 3$, the number of codewords is bounded above as follows:

$$M \leq \frac{2^n}{n+1}.$$

Hamming Codes

- Hamming codes achieve equality

$$M = \frac{2^n}{n+1}.$$

- Requires $n+1$ to divide 2^n - possible only if $(n+1)$ is a power of 2
- Let $n+1 = 2^r$, i.e., $n = 2^r - 1$
- Recall that $M = 2^k$ where k is the number of information bits

$$2^k = \frac{2^n}{2^r} = 2^{n-r} \implies k = n - r = 2^r - r - 1$$

Hamming Codes

- General Hamming codes for $r \geq 2$:
  - $n = 2^r - 1$
  - $k = 2^r - r - 1$
  - $d_{min} = 3$
- Examples: (n, k) Hamming codes
  - $r = 2 \implies (3, 1)$ Hamming code (repetition code)
  - $r = 3 \implies (7, 4)$ Hamming code
  - $r = 4 \implies (15, 11)$ Hamming code

Hamming Bound for General d_{min}

- Define
$$t = \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor$$
- We need non-overlapping Hamming balls of radius t around each codeword
- $M|B(c,t)| \leq 2^n$
- $|B(c,t)| = ?$

Number of Strings in Hamming Balls

- All bit strings which are up to t bit flip away from codeword c
  - Zero bit flips away: original codeword: 1 string
  - One bit flip away: n strings (one out of n bits changes)
  - Two bit flips away: $\binom{n}{2}$ strings
    - Ways to choose which 2 bits flip out of the total n bits
    - Given by

$$\binom{n}{2} = \frac{n!}{2!(n-2)!}$$

  - Three bit flips away: $\binom{n}{3}$ strings
  - And so on...

$$|B(c,t)| = 1 + n + \binom{n}{2} + \binom{n}{3} + \ldots + \binom{n}{t}$$

$$= \sum_{i=0}^{t} \binom{n}{i}$$

General Hamming Bound

Theorem

For any binary code of length n bits with minimum distance d_{min}, the number of codewords M satisfies

$$M \leq \frac{2^n}{\sum_{i=0}^{t} \binom{n}{i}},$$

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

- $\binom{n}{t}$: ways to choose t objects out of a total of n objects
- $\binom{n}{t} = \frac{n!}{t!(n-t)!}$
- $m! = m \times (m-1) \times (m-2) \times \ldots \times 2 \times 1$

Thank You!