

# Error control coding: An introduction to linear block codes

Adrish Banerjee

Department of Electrical Engineering  
Indian Institute of Technology Kanpur  
Kanpur, Uttar Pradesh  
India

Mar. 28, 2016



## Lecture #7A: Bounds on the size of a code



# Outline of the lecture

- Hamming bound

# Outline of the lecture

- Hamming bound
  - Perfect codes

# Outline of the lecture

- Hamming bound
  - Perfect codes
- Singleton bound



# Outline of the lecture

- Hamming bound
  - Perfect codes
- Singleton bound
  - Maximum distance separable codes



# Outline of the lecture

- Hamming bound
  - Perfect codes
- Singleton bound
  - Maximum distance separable codes
- Plotkin Bound



# Outline of the lecture

- Hamming bound
  - Perfect codes
- Singleton bound
  - Maximum distance separable codes
- Plotkin Bound
- Gilbert-Varshamov bound



## Bounds on the size of a code

- The basic problem is to find the largest code of a given length,  $n$  and minimum distance,  $d$ .

## Bounds on the size of a code

- The basic problem is to find the largest code of a given length,  $n$  and minimum distance,  $d$ .
- Let  $A(n, d)$  be the maximum number of codewords in any binary code of length  $n$  and minimum distance  $d$  between the codewords.

## Bounds on the size of a code

- The basic problem is to find the largest code of a given length,  $n$  and minimum distance,  $d$ .
- Let  $A(n, d)$  be the maximum number of codewords in any binary code of length  $n$  and minimum distance  $d$  between the codewords.
- We are interested in finding the maximum number of binary codewords  $A(n, d)$  from the  $n$ -dimensional vector space.

## Bounds on the size of a code

- The basic problem is to find the largest code of a given length,  $n$  and minimum distance,  $d$ .
- Let  $A(n, d)$  be the maximum number of codewords in any binary code of length  $n$  and minimum distance  $d$  between the codewords.
- We are interested in finding the maximum number of binary codewords  $A(n, d)$  from the  $n$ -dimensional vector space.
- In other words, we are interested in finding the minimum parity bits  $(n - k)$  required for a  $t$ -correcting binary code of length  $n$ .

# Hamming Bound

- For any binary  $(n, k)$  linear code with minimum distance  $2t + 1$  or greater, the number of parity-check bits satisfies the following inequality:

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

Proof:



# Hamming Bound

- For any binary  $(n, k)$  linear code with minimum distance  $2t + 1$  or greater, the number of parity-check bits satisfies the following inequality:

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

Proof:

- Recall that all the vectors of weight  $t$  or less can be used as coset leaders.



# Hamming Bound

- For any binary  $(n, k)$  linear code with minimum distance  $2t + 1$  or greater, the number of parity-check bits satisfies the following inequality:

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

Proof:

- Recall that all the vectors of weight  $t$  or less can be used as coset leaders.
- Number of vectors (n-tuple) of weight  $t$  or less are:

$$\binom{n}{0} + \binom{n}{1} + \cdots + \binom{n}{t}$$

Navigation icons: back, forward, search, etc.

# Hamming Bound

- Total number of coset leaders are  $2^{n-k}$ .

Navigation icons: back, forward, search, etc.

# Hamming Bound

- Total number of coset leaders are  $2^{n-k}$ .
- Therefore, we have

$$2^{n-k} \geq \binom{n}{0} + \binom{n}{1} + \cdots + \binom{n}{t}$$

# Hamming Bound

- Total number of coset leaders are  $2^{n-k}$ .
- Therefore, we have

$$2^{n-k} \geq \binom{n}{0} + \binom{n}{1} + \cdots + \binom{n}{t}$$

- Taking logarithm on both sides of the inequality, we get

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

# Perfect code

- A  $t$ —error correcting  $(n, k)$  block code is called a perfect code, if its standard array has all the error patterns of  $t$  or fewer errors and no other error pattern as their coset leaders.



# Perfect code

- A  $t$ —error correcting  $(n, k)$  block code is called a perfect code, if its standard array has all the error patterns of  $t$  or fewer errors and no other error pattern as their coset leaders.
- Perfect code satisfies the Hamming bound with equality.



## Perfect code

- A  $t$ —error correcting  $(n, k)$  block code is called a perfect code, if its standard array has all the error patterns of  $t$  or fewer errors and no other error pattern as their coset leaders.
- Perfect code satisfies the Hamming bound with equality.
- Examples of perfect codes: single error correcting Hamming code, triple error correcting (23,12) Golay code.



## Perfect code

- A  $t$ —error correcting  $(n, k)$  block code is called a perfect code, if its standard array has all the error patterns of  $t$  or fewer errors and no other error pattern as their coset leaders.
- Perfect code satisfies the Hamming bound with equality.
- Examples of perfect codes: single error correcting Hamming code, triple error correcting (23,12) Golay code.
- Note perfect codes are not the best error correcting codes.



# Singleton Bound

- The minimum distance  $d_{\min}$  of an  $(n, k)$  linear code satisfies the following inequality

$$d_{\min} \leq n - k + 1$$

Proof:



# Singleton Bound

- The minimum distance  $d_{\min}$  of an  $(n, k)$  linear code satisfies the following inequality

$$d_{\min} \leq n - k + 1$$

Proof:

- For an  $(n, k)$  code that an  $(n - k) \times n$  parity check matrix,  $\mathbf{H}$ , the row rank of any  $\mathbf{H}$  is  $(n-k)$ .



# Singleton Bound

- The minimum distance  $d_{\min}$  of an  $(n, k)$  linear code satisfies the following inequality

$$d_{\min} \leq n - k + 1$$

Proof:

- For an  $(n, k)$  code that an  $(n - k) \times n$  parity check matrix,  $\mathbf{H}$ , the row rank of any  $\mathbf{H}$  is  $(n-k)$ .
- Hence, the column rank of any  $\mathbf{H}$  is  $(n-k)$ . Any combinations of  $(n-k+1)$  columns of  $\mathbf{H}$  must be linearly dependent.



# Singleton Bound

- The minimum distance  $d_{\min}$  of an  $(n, k)$  linear code satisfies the following inequality

$$d_{\min} \leq n - k + 1$$

Proof:

- For an  $(n, k)$  code that an  $(n - k) \times n$  parity check matrix,  $\mathbf{H}$ , the row rank of any  $\mathbf{H}$  is  $(n-k)$ .
- Hence, the column rank of any  $\mathbf{H}$  is  $(n-k)$ . Any combinations of  $(n-k+1)$  columns of  $\mathbf{H}$  must be linearly dependent.
- Recall, that the minimum distance of a code is equal to the minimum number of nonzero columns in  $\mathbf{H}$  that are linearly dependent.



# Singleton Bound

- The minimum distance  $d_{\min}$  of an  $(n, k)$  linear code satisfies the following inequality

$$d_{\min} \leq n - k + 1$$

Proof:

- For an  $(n, k)$  code that an  $(n - k) \times n$  parity check matrix,  $\mathbf{H}$ , the row rank of any  $\mathbf{H}$  is  $(n-k)$ .
- Hence, the column rank of any  $\mathbf{H}$  is  $(n-k)$ . Any combinations of  $(n-k+1)$  columns of  $\mathbf{H}$  must be linearly dependent.
- Recall, that the minimum distance of a code is equal to the minimum number of nonzero columns in  $\mathbf{H}$  that are linearly dependent.
- Hence,

$$d_{\min} \leq n - k + 1$$



# Singleton Bound

Another proof:

- Any nonzero codeword with only one information weight can at most have  $n - k + 1$  codeword weight.



# Singleton Bound

Another proof:

- Any nonzero codeword with only one information weight can at most have  $n - k + 1$  codeword weight.
- Since, minimum distance of a code is equal to the minimum weight of a nonzero codeword.

$$d_{\min} \leq n - k + 1$$



# Maximum Distance Separable (MDS) Codes

- MDS codes satisfies the Singleton bound with equality, i.e.

$$n - k = d - 1$$



# Maximum Distance Separable (MDS) Codes

- MDS codes satisfies the Singleton bound with equality, i.e.

$$n - k = d - 1$$

- MDS code has the maximum possible distance between the codewords.



# Maximum Distance Separable (MDS) Codes

- MDS codes satisfies the Singleton bound with equality, i.e.

$$n - k = d - 1$$

- MDS code has the maximum possible distance between the codewords.
- Example: Repetition codes, Single parity check codes, Reed-Solomon codes.



# Plotkin Bound

- The minimum distance  $d_{\min}$  of an  $(n, k)$  linear code satisfies the following inequality

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

Proof:



# Plotkin Bound

- The minimum distance  $d_{\min}$  of an  $(n, k)$  linear code satisfies the following inequality

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

Proof:

- Consider an  $(n, k)$  linear code  $C$  with generator matrix  $\mathbf{G}$ . Arrange the  $2^k$  codewords of  $C$  as a  $2^k \times n$  array.



# Plotkin Bound

- The minimum distance  $d_{\min}$  of an  $(n, k)$  linear code satisfies the following inequality

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

Proof:

- Consider an  $(n, k)$  linear code  $C$  with generator matrix  $\mathbf{G}$ . Arrange the  $2^k$  codewords of  $C$  as a  $2^k \times n$  array.
- Each column of this array has  $2^{k-1}$  zeros and  $2^{k-1}$  ones.



# Plotkin Bound

- The minimum distance  $d_{\min}$  of an  $(n, k)$  linear code satisfies the following inequality

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

Proof:

- Consider an  $(n, k)$  linear code  $C$  with generator matrix  $\mathbf{G}$ . Arrange the  $2^k$  codewords of  $C$  as a  $2^k \times n$  array.
- Each column of this array has  $2^{k-1}$  zeros and  $2^{k-1}$  ones.
  - Show that the number of codewords that “1” at the  $l$ -th position is same as number of codewords that have “0” at the  $l$ -th position.



# Plotkin Bound

Proof (contd.):

- In the code array, each column contains at least one nonzero entry.



# Plotkin Bound

Proof (contd.):

- In the code array, each column contains at least one nonzero entry.
- Consider the  $l$ -th column of the code array. Let  $S_0$  be the codewords with a “0” at the  $l$ -th position and  $S_1$  be the codewords with a “1” at the  $l$ -th position. Let  $\mathbf{x}$  be a codeword from  $S_1$ .



# Plotkin Bound

Proof (contd.):

- In the code array, each column contains at least one nonzero entry.
- Consider the  $l$ -th column of the code array. Let  $S_0$  be the codewords with a “0” at the  $l$ -th position and  $S_1$  be the codewords with a “1” at the  $l$ -th position. Let  $\mathbf{x}$  be a codeword from  $S_1$ .
- Adding  $\mathbf{x}$  to each vector in  $S_0$ , we obtain a set  $S'_1$  of codewords with a “1” at the  $l$ -th position.

$$|S'_1| = |S_0| \quad \text{and} \quad S'_1 \subseteq S_1$$



# Plotkin Bound

Proof (contd.):

- In the code array, each column contains at least one nonzero entry.
- Consider the  $l$ -th column of the code array. Let  $S_0$  be the codewords with a “0” at the  $l$ -th position and  $S_1$  be the codewords with a “1” at the  $l$ -th position. Let  $\mathbf{x}$  be a codeword from  $S_1$ .
- Adding  $\mathbf{x}$  to each vector in  $S_0$ , we obtain a set  $S'_1$  of codewords with a “1” at the  $l$ -th position.

$$|S'_1| = |S_0| \quad \text{and} \quad S'_1 \subseteq S_1$$

- The above condition implies that

$$|S_0| \leq |S_1| \tag{1}$$



# Plotkin Bound

Proof (contd.):

- Adding  $\mathbf{x}$  to each vector in  $S_1$ , we obtain a set  $S'_0$  of codewords with a "0" at the  $l$ -th position.

$$|S'_0| = |S_1| \quad \text{and} \quad S'_0 \subseteq S_0$$

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↺ 🔍 ↻

# Plotkin Bound

Proof (contd.):

- Adding  $\mathbf{x}$  to each vector in  $S_1$ , we obtain a set  $S'_0$  of codewords with a "0" at the  $l$ -th position.

$$|S'_0| = |S_1| \quad \text{and} \quad S'_0 \subseteq S_0$$

- - The above condition implies that

$$|S_1| \leq |S_0| \tag{2}$$

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↺ 🔍 ↻

# Plotkin Bound

Proof (contd.):

- Adding  $\mathbf{x}$  to each vector in  $S_1$ , we obtain a set  $S'_0$  of codewords with a "0" at the  $l$ -th position.

$$|S'_0| = |S_1| \quad \text{and} \quad S'_0 \subseteq S_0$$

- - The above condition implies that

$$|S_1| \leq |S_0| \tag{2}$$

- From (1) and (2), we get  $|S_0| = |S_1|$ . Therefore  $l$ -th column contains  $2^{k-1}$  zeros and  $2^{k-1}$  ones.

# Plotkin Bound

Proof (contd.):

- Adding  $\mathbf{x}$  to each vector in  $S_1$ , we obtain a set  $S'_0$  of codewords with a "0" at the  $l$ -th position.

$$|S'_0| = |S_1| \quad \text{and} \quad S'_0 \subseteq S_0$$

- - The above condition implies that

$$|S_1| \leq |S_0| \tag{2}$$

- From (1) and (2), we get  $|S_0| = |S_1|$ . Therefore  $l$ -th column contains  $2^{k-1}$  zeros and  $2^{k-1}$  ones.

- Thus the total number of ones in the array is  $n \cdot 2^{k-1}$ .

# Plotkin Bound

- Each nonzero codeword has weight atleast  $d_{\min}$ . Hence,

$$(2^k - 1) \cdot d_{\min} \leq n \cdot 2^{k-1}$$



# Plotkin Bound

- Each nonzero codeword has weight atleast  $d_{\min}$ . Hence,

$$(2^k - 1) \cdot d_{\min} \leq n \cdot 2^{k-1}$$

- This implies that

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



# Gilbert-Varshamov Bound

- There exists an  $(n, k)$  linear code with a minimum distance of at least  $d$  that satisfies the following inequality

$$1 + \binom{n-1}{1} + \cdots + \binom{n-1}{d-2} < 2^{n-k}$$

Proof:



# Gilbert-Varshamov Bound

- There exists an  $(n, k)$  linear code with a minimum distance of at least  $d$  that satisfies the following inequality

$$1 + \binom{n-1}{1} + \cdots + \binom{n-1}{d-2} < 2^{n-k}$$

Proof:

- We shall construct an  $n - k \times n$  parity check matrix,  $\mathbf{H}$  with the property that no  $d - 1$  columns are linearly dependent.



# Gilbert-Varshamov Bound

- There exists an  $(n, k)$  linear code with a minimum distance of at least  $d$  that satisfies the following inequality

$$1 + \binom{n-1}{1} + \cdots + \binom{n-1}{d-2} < 2^{n-k}$$

Proof:

- We shall construct an  $(n-k) \times n$  parity check matrix,  $\mathbf{H}$  with the property that no  $d-1$  columns are linearly dependent.
- Recall, that this will ensure a minimum distance of  $d$ .



# Gilbert-Varshamov Bound

- There exists an  $(n, k)$  linear code with a minimum distance of at least  $d$  that satisfies the following inequality

$$1 + \binom{n-1}{1} + \cdots + \binom{n-1}{d-2} < 2^{n-k}$$

Proof:

- We shall construct an  $(n-k) \times n$  parity check matrix,  $\mathbf{H}$  with the property that no  $d-1$  columns are linearly dependent.
- Recall, that this will ensure a minimum distance of  $d$ .
- The first column could be any nonzero  $(n-k)$ -tuple.



# Gilbert-Varshamov Bound

- There exists an  $(n, k)$  linear code with a minimum distance of at least  $d$  that satisfies the following inequality

$$1 + \binom{n-1}{1} + \cdots + \binom{n-1}{d-2} < 2^{n-k}$$

Proof:

- We shall construct an  $(n-k) \times n$  parity check matrix,  $\mathbf{H}$  with the property that no  $d-1$  columns are linearly dependent.
- Recall, that this will ensure a minimum distance of  $d$ .
- The first column could be any nonzero  $(n-k)$ -tuple.
- Suppose we have chosen  $i$  columns so that no  $d-1$  columns are linearly dependent.



# Gilbert-Varshamov Bound

- Maximum number of distinct linear combinations of these  $i$  columns taken  $d-2$  or fewer at a time is given by  $N_i$

$$N_i = \binom{i}{1} + \cdots + \binom{i}{d-2}$$



# Gilbert-Varshamov Bound

- Maximum number of distinct linear combinations of these  $i$  columns taken  $d - 2$  or fewer at a time is given by  $N_i$

$$N_i = \binom{i}{1} + \cdots + \binom{i}{d-2}$$

- If this number,  $N_i$  is less than all possible nonzero  $n - k$ -tuple, i.e.  $2^{n-k} - 1$ , we can add another column different from these linear combinations, and keep the property that any  $d - 1$  columns of the new  $(n - k) \times (i + 1)$  array are linearly independent.



# Gilbert-Varshamov Bound

- Maximum number of distinct linear combinations of these  $i$  columns taken  $d - 2$  or fewer at a time is given by  $N_i$

$$N_i = \binom{i}{1} + \cdots + \binom{i}{d-2}$$

- If this number,  $N_i$  is less than all possible nonzero  $n - k$ -tuple, i.e.  $2^{n-k} - 1$ , we can add another column different from these linear combinations, and keep the property that any  $d - 1$  columns of the new  $(n - k) \times (i + 1)$  array are linearly independent.
- We continue doing this as long as the following condition is satisfied.

$$\binom{i}{1} + \cdots + \binom{i}{d-2} < 2^{n-k} - 1$$



# Gilbert-Varshamov Bound

- Maximum number of distinct linear combinations of these  $i$  columns taken  $d - 2$  or fewer at a time is given by  $N_i$

$$N_i = \binom{i}{1} + \cdots + \binom{i}{d-2}$$

- If this number,  $N_i$  is less than all possible nonzero  $n - k$ -tuple, i.e.  $2^{n-k} - 1$ , we can add another column different from these linear combinations, and keep the property that any  $d - 1$  columns of the new  $(n - k) \times (i + 1)$  array are linearly independent.
- We continue doing this as long as the following condition is satisfied.

$$\binom{i}{1} + \cdots + \binom{i}{d-2} < 2^{n-k} - 1$$

- The above condition should hold for all  $n$  columns of the parity check matrix,  $\mathbf{H}$ .

