International Journal of Innovative Research in Science and Engineering
Vol. No.3, Issue 03, March 2017
www.ijirse.com
ISSN (O) 2454-9665
ISSN (P) 2454-0663

AN INTRODUCTION TO THE THEORY OF ERROR-CORRECTING CODES

Ritu Ahuja

[1]*Khalsa College for Women, Civil Lines, Ludhiana, Punjab (India)*

ABSTRACT

Algebraic coding theory or coding theory is the applied part of mathematics which deals with the design of error-correcting codes so that reliable transmission of information is possible across noisy channels. All real systems that work with digitally represented data requires to use error correcting codes as all real channels are to some extent, noisy. Coding theory problems are therefore among the very basic and most frequent problems of storage and transmission of information. It is the key to another area of study, Information Theory, which lies in the intersection of probability and coding theory. Its methods involve the use of both classical and modern algebraic techniques involving finite fields, group theory and polynomial algebra. This paper presents the basic concepts and results of coding theory.

Keywords: Coding theory, Error-correcting codes, Noisy channels.

I. INTRODUCTION

Coding theory and information theory are the twin disciplines which originated in late 1940's with Claude Shannon's 1948 paper 'A Mathematical Theory Of communication'. Coding theory developed to become a part of mathematics. Information passes from a source to sink via a channel. The uncertainties of channel may distorts or damage the information in the passage. Coding theory attempts to detect and correct errors using algebraic means. Though there are many codes but linear codes are the most common. We shall examine linear codes and their properties.

II. LINEAR CODES

Linear codes are some of the most basic codes in coding theory, as they possess an algebraic structure, are simple to understand and effective for practical applications. They are used to encode messages that are then sent over noisy communication channels. Data is encoded by adding certain amount of redundancy so that original message can be recovered if errors have occurred. A q-ary code is a code over an alphabet {0,1,….,q-1} .The alphabet is taken to be Galois field of order q, usually denoted as GF(q). For convenience we shall consider binary codes (i.e. a sequence of 0's and 1's). However, properties and theorems of linear codes still hold true for other number bases.

Consider the message u = u1u2 : : : uk, where each *ui* is represented by a 0 or 1.

We let x = *x1x2 : : : xn* be a function of the message u such that $n \geq k$.

Definition 1.1. A *code* is a set X such that for all x ε X, x is a code word.

ISSN (O) 2454-9665
ISSN (P) 2454-0663

***Definition* 1.2.** An *error correcting code* is an algorithm for expressing a sequence of numbers such that any errors which are introduced can be detected and corrected (within certain limitations) based on the remaining numbers.

Let V(n,q) denotes the vector space of n tuples over a Galois field GF(q) of order q (taken as a power of prime)

***Definition* 1.3.** A [n,k] linear code over a finite field of order q is just a k-dimensional subspace of V(n,q).

So,a subset C of V(n,q) is a linear code iff u+v , au ε GF(q).In particular, a binary code is linear if and only if the sum of any two codeworfs is a codeword.

Example1.4. C= {000,011,101,110} is a [3,2] linear code.

***Definition* 1.4.** A kxn matrix G whose rows form a basis of linear [n,k] code is called generator matrix of the code.

G can be transformed to the standard form $[I_k, A]$ where I_k is the kxk identity matrix and A is a kxn matrix.

***Encoding*:** Messages can be identified with q tuples of V(k,q). A message vector $u=u_1u_2...u_k$ is encoded by post multiplying it by G.

$$X = uG = x_1\, x_2....x_k\, x_{k+1}.....x_n$$

If G is in standared form, first K symbols of code word represent the message itself:

$x_1 = u_1$, $x_2 = u_2$……$x_k=u_k$.

The next n-k symbols are *check symbols*.

***Definition* 1.5.** The (n-k) x n matrix H whose rows are the *parity checks* on the code words is called parity check matrix.

Properties of liner codes – After defining the liner codes here are some properties of such codes.

1. Given parity check matrix H, $Hx^T = 0$ for all x ε C
   So, C= {x ε V(n,q)} / $Hx^T = 0$} Hence, Parity check matrix completely specifies a linear code.
2. For the standared form of generator matrix G, parity check matrix H = $[-A^T\, I_{n-k}]$.
3. Furthermore, $GH^T=0$ and $HG^T=0$.
4. An [n,k] linear code has efficiency or rate = k/n
5. Given x and y are code words, $H(x+y)^T= Hx^T+ Hy^T= 0$ implies x+y is also a code word, the property defining the linearity of x.

   *Example*1.6. let H be the parity check matrix given by

$$H = \begin{bmatrix} 1\,0\,1\,1\,0\,0 \\ 0\,1\,1\,0\,1\,0 \\ 1\,0\,0\,0\,0\,1 \end{bmatrix}$$

Its order is 3x6 so the associated code has length 6 and dimension 3: it is a [6,3] code. First 3 symbols being the original message itself, $x_1 = u_1$, $x_2 = u_2$, $x_3=u_3$ so using $Hx^T = 0$ and substituting relevant u_1,

$u_1+ u_3+ x_4 = 0$

$u_2+ u_3+ x_5 = 0$

$u_1+x_6 = 0$

For given message $u = u_1\, u_2u_3$, the code word x can be created using these eqs. For instance, if x = 101

$x_4 = 0$, $x_5 = 1$, $x_6 = 1$

International Journal of Innovative Research in Science and Engineering
Vol. No.3, Issue 03, March 2017
www.ijirse.com
ISSN (O) 2454-9665
ISSN (P) 2454-0663

Code word is x = 101011

Error Detection: Linear codes allow for well-defined algorithm for error detection. How well an error is detected is based on code's Hamming weight.

***Definition*1.7.** The *Hamming distance* between two vectors x = *x*1*x*2 *: : : xn* and y = *y*1*y*2 *: : : yn* is denoted by *dist*(*x; y*) and is equal to the number of places where they differ. For example,

dist(100111*;* 111011) = 3*; dist*(101010*;* 010110) = 4*:*

***Definition*1.8.** The *Hamming weight* of a vector x = *x*1*x*2 *: : : xn* is the numberof *xi* ≠0, and is written as *wt*(u). For example, *wt*(100111) = 4*; wt*(101010) = 3*:*

***Definition* 1.9**. Given a code X, the *minimum distance* of the code is the minimum Hamming distance between its codewords. It is also known as simply the *distance* of the code.

$$d(x, y) = \min \{d(x, y) \mid x,y \in C, x \neq y\}$$

From this, it can be deduced that $d(x, y) = wt(x - y)$ and $min\ d(x; y) = min\ wt(x - y)$

***Theorem* 1.10. (Basic error correcting theorem)**

1. A code C can detect upto S errors If $d(x, y) \geq S+1$
2. A code C can correct upto t errors if $d(x, y) \geq 2t+1$

PF. 1. Trivial

2. Suppose $dist(x, y) \geq 2t + 1$. Let a codeword x is transmitted and a word y is received with $d(x, y) \leq t$. If $x' \neq x$ is a codeword, then $d(x, y) \geq t + 1$ because otherwise $d(x', y) < t + 1$ and therefore $d(x, x') \leq d(x, y) + d(y, x') < 2t + 1$ what contradicts the assumption $d(C) \geq 2t + 1$.

Decoding – Once the message u has been encoded into code word x and sent through communication channel, decoding process can take place. Due to channel noise, receiver may not receive x as the coded message, say the received vector is y.

The vector $e = y\text{-}x = e_1\ e_2 \ldots\ldots e_n$ is the error vector. In order to decode the received message y, it suffices to know e, as x= y-e. One must pick the error that is most likely to occur. Since any codeword x is equally likely to occur, this method minimizes the probability of the decoder making a mistake and is called the *maximum likelihood decoding* (MLD)

In *Nearest Neighbor Decoding*, y is received and x is chosen to minimize the hamming distance $d(x, y) = \{i : x_i \neq y_i\}$. In other words, this method choses the y ε C, that is closest to x ε V(n,2). As with the other methods, in the event that more than one distinct y is chosen, the message is retransmitted in the hope the errors will not occur in the same fashion.

***Definition*1.11.** A *binary symmetric channel* is a channel with binary inputs and binary outputs and error probability *p*. In a binary symmetric channel, given the error vector $e = e_1e_2 : : : e_n$, the probability that $e_i = 0$ is 1-*p* whereas the probability that $e_i = 1$ is *p* where generally,$0 \leq p \leq ½$.In general, the following equation holds true for the probability of the occurrence e, Pr(e), given an fixed vector w of weight *a*. $Pr(e = w) = p^a(1\text{-}p)^{n-a}$

Given that $p < 1/2$, it follows that $(1- p) > p$, and

$(1 - p)^n > p(1 - p)^{n-1} > p^2(1 -p)^{n-2} > \ldots\ldots$

International Journal of Innovative Research in Science and Engineering
Vol. No.3, Issue 03, March 2017
www.ijirse.com
ISSN (O) 2454-9665
ISSN (P) 2454-0663

Therefore, the error vector with the least weight is most likely, so the decoder picks that vector as the error vector. This method is also known as *nearest neighbor decoding*. From here, a brute force plan is required, as each received codeword y is compared to all 2^k possible codewords x using the chosen error vector. However,this method is near impossible for *k* large. Therefore, there must be another method that can decode the message faster. Another method used to decode codes is using a *standard array (or Slepian array).*For this method, we must give a definition.

Definition 1.12. Given the set X of all linear codes x, for any vector a, the set

$$a + X \{ a+x \mid x \varepsilon X\}$$

is called a *coset* of x.

Given the received vector y, y must belong to some coset. Suppose, $y = a_i + x$,then for the sent codeword x' , the error vector would be $e = y - x' = a_i + x - x' = a_i + x''$, which is contained in the set $a_i + X$. Therefore, all the possible error vectors are the vectors in the coset containing y. The decoder's approach now becomes choosing the minimum weight vector ê in this coset; this vector is called the *coset leader*. Now, y can be decoded as ^x = y - ê. We let $\{a_i\}$ be the set of coset leaders. We find the coset leaders in a *standard array*. The standard array is a table, with the first row consisting of the possible original messages u, the second row consisting of the codewords x with the 0 codeword as the first element of the row, and the other rows consists of the other cosets $a_i + X$ with coset leader (error vector) as the first element of each row.

Row 1 =	u_1	u_2	---------	u_2^k
Row 2 =	x_1	x_2	---------	x_2^k
Row 3 =	a_1+x_1	a_1+x_2	---------	$a_1+x_2^k$
	.			
	.			
	.			
Row n+2 =	a_n+x_1	a_n+x_2	---------	$a_n+x_2^k$

On receiving y, its position in array is found .Decoder decides that error vector e is the coset leader found at the extreme left of y and x follows from y. Briefly, a received vector is decoded as the codeword at the top of its column in the standard array. Since coset leader is the minimum weight vector in each coset, *standard array decoding* is *a nearest neighbor decoding scheme*

*Example*1.13. Let C be the binary [5,2] code with generator matrix

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

Hence, the code C={00000,10110,01011,11101}

A standard array for the code can be created as follows

International Journal of Innovative Research in Science and Engineering
Vol. No.3, Issue 03, March 2017
www.ijirse.com
ISSN (O) 2454-9665
ISSN (P) 2454-0663

	0010	01	11	
Codewords →	00000	10110	01011	11101
	10000	00110	01011	01101
	01000	11110	00011	10101
	00100	10010	01111	11001
	00010	10100	01001	11111
	00001	10111	01010	11100
	11000	01110	10011	00101
	01100	11010	00111	10001
	↑			
	cosetleader			

Using this array,any received vector y can be decoded to the most lokely codeword x using the equation X=y-e. since error vectors are the first elements in each row, the codeword x corresponding to y is the second element of the codeword at the top of column containing y in the standard array.

So, if the received vector is y=11111, then x=11101, and u=11

The method of Standard array in practice can be a long process if n is large. There are other methods like syndrome decoding.

Definition 1.14.. Given the received codeword y and parity check matrix *H*, the *syndrome* of y is

$$S = HyT$$

The syndrome identifies errors in the received codeword. The value of the syndrome is the position of the code where the error is. With a binary code, this also implies that the error can be easily corrected. The syndrome tells us the erroneous symbol of the code .With a binary code, if the erroneous symbol is 0, we simply switch it to a 1, and vice versa. Then what is left is the sent codeword which is easy to decode. This is the *syndrome decoding method*.

One may use the decoding technique suited best to the situation.

2.The Sphere Packing or (Hamming Bound)

Lemma 2.1 Suppose x is a vector of length n and that k is a nonnegative integer not exceeding n. Then there are C(n, 0) + C(n, 1) + · · · + C(n, k)

vectors y of length n such that d(x, y) ≤ k (where d is the Hamming distance)

Proof: Let i be a nonnegative integer. The number of vectors y with d(x, y) = i equals the number of ways to select the i locations where x and y differ. This can be done in C(n, i) ways. It follows that there are C(n, 0) + C(n, 1) + · · · + C(n, k) vectors such that d(x, y) ≤ k.

It can also be understood by geometric means. By the sphere of radius k centered at x we mean the set of all n-tuples y such that d(x, y) ≤ k. Lemma 1 says that there are exactly $\sum_{i=0}^{k} C(n, i)$ vectors in the sphere of radius k centered at x.

Lemma 2.2 Let C be a binary code containing codewords of length n and let d(C) = 2k + 1. Then given a vector y of length n, there is at most one codeword x such that y is in the sphere of radius k centered at x.

Proof: Suppose that y is in the sphere of radius k centered at two different codewords x_1 and x_2. Then $d(x_1, y) \leq k$ and $d(x_2, y) \leq k$. By the triangle inequality for the Hamming distance this implies that

$d(x_1, x_2) \leq d(x_1, y) + d(x_2, y) \leq k + k = 2k$, contradicting the fact that the minimum distance between codewords is 2k+1.

ISSN (O) 2454-9665
ISSN (P) 2454-0663

A useful bound on how many codewords can be in a code consisting of n-tuples that can correct a specified number of errors is the sphere packing bound.

Theorem2.3 *The Sphere Packing or (Hamming Bound)*

Suppose that C is a code of length n with d(C) =2k +1. Then M, the number of codewords in C, cannot exceed

$$2^n / C(n, 0) + C(n, 1) + \cdots + C(n, k).$$

Proof: There are 2^n vectors of length n. By Lemma 1, the sphere of radius k centered at a codeword x contains $C(n, 0) + C(n, 1) + \cdots + C(n, k)$ vectors. Since no vector can be in two such spheres (by Lemma 2), the number of vectors of length n is at least as large as the number of codewords

times the number of vectors in each such sphere. Hence,

$2n \geq M [C(n, 0) + C(n, 1) + \cdots + C(n, k)]$.

Suppose that C is a code of length n with d(C) = 2k+1 Then M, the number of codewords in C, cannot exceed

$2^n / C(n, 0) + C(n, 1) + \cdots + C(n, k)$.

Proof: There are 2^n bit strings of length n. The sphere of radius k centered at a codeword x contains $C(n, 0) + C(n, 1) + \cdots + C(n, k)$ vectors. Since no two codewords can be in two such spheres, the number of bit strings of length n is at least as large as the number of codewords

times the number of bit strings in each such sphere. Hence,

$2^n \geq M [C(n, 0) + C(n, 1) + \cdots + C(n, k)]$.

A (n, M, d) code has M codewords of length n each with minimum distance d.

Main coding theory problem

A good (n, M, d)-code should have

- small n, for fast transmission of messages,
- large M, to enable transmission of wide variety of messages and
- Large d, to correct many errors.

However, these aims are conflicting. One or other parameters has to be compromised to achieve optimum value of the desired parameter. The *Main coding theory problem* is to optimize one of the parameters n, m, d for the given values of other two. A usual version of the problem is to find the largest code of given length and given minimum distance.

REFERENCES

[1] F.J. MacWilliams and N.J.A Sloane, the theory of error-correcting codes (North-Holland Publishing Company. 1978).

[2] R. Hill, a first course in coding theory (Oxford University Press, 1986).

[3] R.E. Blahut, theory and practice of error control codes (Addison-Wesley, 1983).

[4] Vera Pless, introduction to the theory of error-correcting codes (Third Edition. John Wiley and Sons, Inc. 1998).

[5] Van lint, Introduction to coding theory, (Third edition, Graduate texts in Mathematics, Springer 1998).