I

Stephen Lucci - Summer 2010

Algebraic Coding Theory

E ** l * sh co * ta * n * m * ch red * nd * ncy

It is this redundancy that is the basis for error detection and error correction in digital transmission (and storage).

Some causes for errors:

- thermal noise
- interference
- cross talk
- packet loss

We desire error rates of one bit per 10^{12} bits. (per trillion).

"prevention of errors" vs. "error correcting code"

A few discrete communication channels:

- microwave links
- coaxial cables
- telephone circuits
- magnetic and optical disks

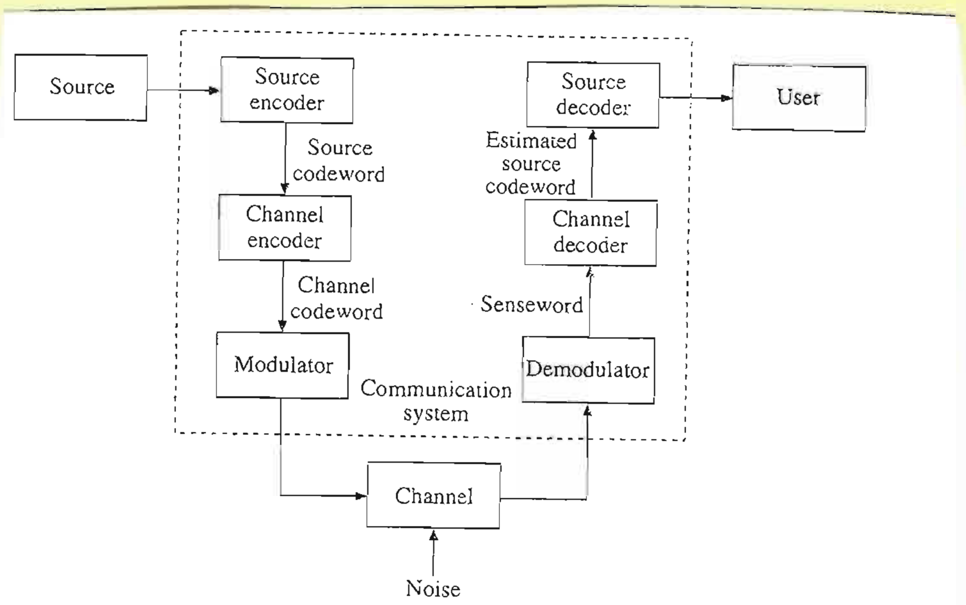
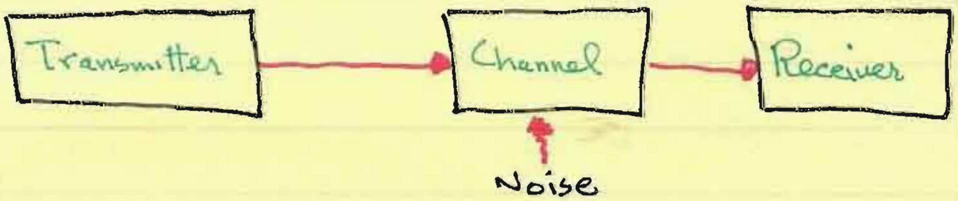
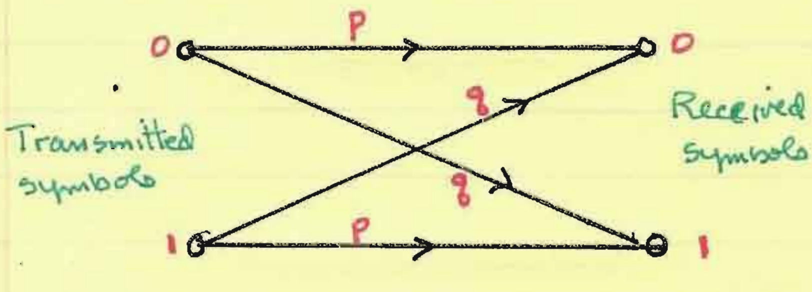


Figure 1.1. Block diagram of a digital communication system

Figure from Algebraic Codes for Data Transmission by Blahut, Cambridge 2006.

With less detail we have:





Binary symmetric channel

p: probability symbol transmitted correctly

$q = 1 - p$: probability of an error due to noise

We assume (for now) that errors occur independently

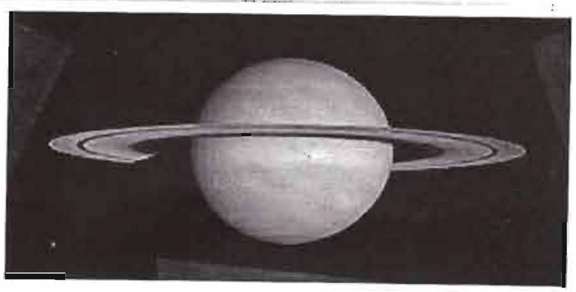
For n binary digits, the probability of r errors is:

$$P(r \text{ errors}) = \binom{n}{r} p^{n-r} q^{r}$$

Early 1980's - Voyager spacecrafts
Sent images of the planet Saturn back to earth.

Channel - the nearly 800,000,000 miles of space separating our two planets

Saturn ♄



Saturn, as seen by Cassini

Each image : 800 × 800 pixels
Each pixel : $2^8 = 256$ degrees of brightness (black & white picture)

Color photo transmitted 3 times

$3 \times 800 \times 800 \times 8 = 15{,}360{,}000$ bits

Brief History of Data transmission codes

1948 - Claude Shannon - Associated with any communication (or storage) channel is a number C (in bits/sec.) called the capacity of the channel

Information transmission rate R (in bits/sec)
If $R < C$
We can design a data transmission code.
Probability of output error through this channel is as small as desired.

1950 - Hamming - single-error-correcting block codes
1954 - Muller - multiple-error-correcting codes
1954 - Reed - A decoding algorithm for Muller's codes
1959 - Hocquenghem
1960 - Bose and Ray-Chaudhuri } large class of multiple e.c. codes (BCH codes)

1980's - compact disks use Reed-Solomon code for correcting double byte errors.
also used in magnetic tape drives
network modems
digital video disks.

A code - adds extra check symbols to data symbols thereby enabling error location and correction.

A binary code of size M and blocklength n

$$\text{binary code} \quad C = \left\{ \begin{array}{c} 10101 \\ 10010 \\ 01110 \\ 11111 \end{array} \right\} \qquad \begin{array}{c} M = 4 \\ n = 5 \end{array}$$

message	codeword
00 →	10101
01 →	10010
10 →	01110
11 →	11111

If 10101 received, assume 00 sent

What if 10011 received? ...
We will assume 01 was sent. ... Why?

I

Definition 1.4.1. *A block code of size M over an alphabet with q symbols is a set of M q-ary sequences of length n called* codewords.

If $q = 2$, the symbols are called bits. Usually, $M = q^k$ for some integer k, and we shall be interested only in this case, calling the code an (n, k) code. Each sequence of k q-ary data symbols can be associated with a sequence of n q-ary symbols comprising a codeword.

There are two basic classes of codes: *block codes* and *trellis codes*. These are illustrated in Figure 1.2. A block code represents a block of k data symbols by an n-symbol codeword. The rate R of a block code[1] is defined as $R = k/n$. Initially, we shall restrict our attention to block codes.

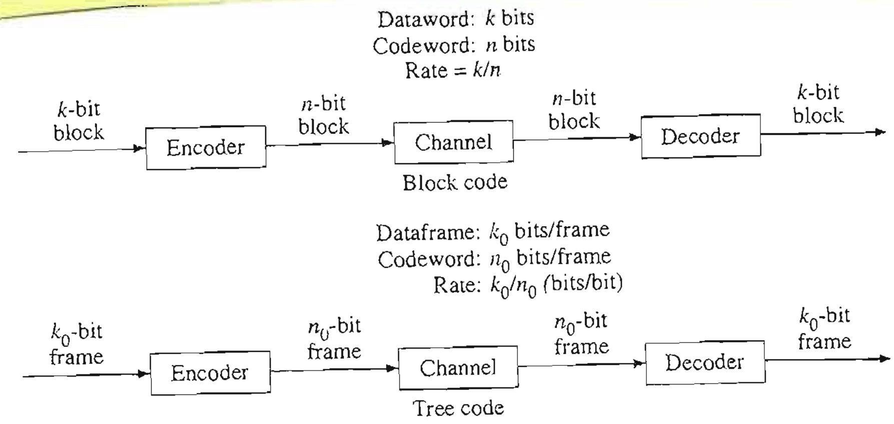


Figure 1.2. Basic classes of codes

Figure from Blahut.

Def. 1.4.2. The Hamming distance $d(x,y)$ between two q-ary sequences x and y of length n is the number of places in which x and y differ.

e.g. $x = 000$, $y = 111$ $\quad d(x,y) = 3$

$x = 011$, $y = 101$ $\quad d(x,y) = 2$

x, y need not be binary: $x = 1231$, $y = 2132$ $\quad d(x,y) = 3$

$d(x,y) \geq 0$ non negative

$d(x,y) = d(y,x)$ symmetric.

$d(x,y) \leq d(x,z) + d(y,z)$ triangle inequality

Def 1.4.3. Let $C = \{c_\ell \mid \ell = 0, \dots, M-1\}$ be a code. The minimum Hamming distance, d_{min} (or d) of C is the Hamming distance between the pair of codewords with smallest Hamming distance. That is:

$$d_{min} = \min_{\substack{c_i, c_j \in C \\ i \neq j}} d(c_i, c_j)$$

In prior code $C = \left\{ \begin{matrix} 10101 \\ 00010 \\ 01110 \\ 11111 \end{matrix} \right\}$ verify that $d_{min} = 2$.

Some elementary codes

Parity-check codes – K data bits
Add $(K+1)^{st}$ bit so that the number of one's in each codeword is even.

example

if $K = 3$

even parity

$$\left.\begin{matrix} 000 & \longleftrightarrow & 0000 \\ 001 & \longleftrightarrow & 0011 \\ 010 & \longleftrightarrow & 0101 \\ & \vdots & \end{matrix}\right\} \begin{matrix} (K+1, K) \\ \text{or} \\ (n, n-1) \text{ code} \end{matrix}$$

parity bit

$d_{min} = 2.$

Similarly one can design an odd parity code with $K=3$ once again

$000 \longleftrightarrow 0001 \quad \dots$

Repetition Codes.

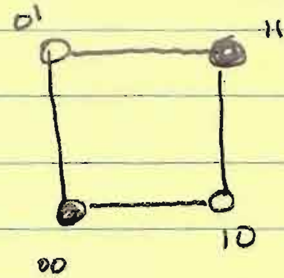
Single error detection code (s.e.d.)

$0 \leftrightarrow 00$

$1 \leftrightarrow 11$

Encoding function E

$E: \mathbb{Z}_2^1 \to \mathbb{Z}_2^2$ where $E(0) = 00$
$E(1) = 11$



Decoding function : D

$D: \mathbb{Z}_2^2 \to \mathbb{Z}_2^1$

$D(00) = 0$

$D(11) = 1$

however

$D(01) = ?$

$D(10) = ?$

Observe that $d(00, 01) = d(11, 01) = 1$

and $d(00, 10) = d(11, 10) = 1$

In each case, we have detected a single error. Error correction is not possible.

Thm: To detect t or fewer errors, the minimum Hamming distance (d_{min}) of a code must be $\geq t+1$.

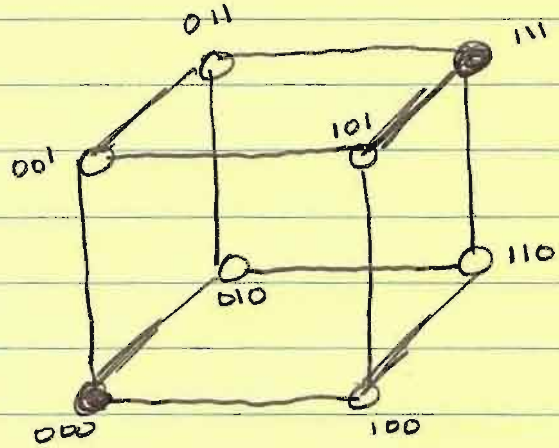
Observe, this repetition code is s.e.d. and $d_{min} = 2$.

Single error correction code (s.e.c.)

Encoding function : E

$E: \mathbb{Z}_2^1 \to \mathbb{Z}_2^3$

$E(0) = 000$
$E(1) = 111$



Decoding function: D

$D: \mathbb{Z}_2^3 \to \mathbb{Z}_2^1$

$D(000) = 0$
$D(111) = 1$

error detected and corrected!
$D(001) = D(010) = D(100) = 0$ ... why?
$D(011) = D(101) = D(110) = 1$ ...

What happens if two errors occur?

Maximum likelihood decoding criterion

Fewer errors are more likely than more errors.

The above repetition code is s.e.c.

→ What has happened to the transmission rate R?
... There must be a better way!

I

Hamming Codes

Hamming codes are single error correcting.

For each m, there is a $(2^m - 1,\ 2^m - 1 - m)$ binary Hamming code.

for $m=3$, $(2^3-1,\ 2^3-1-3) = (7,4)$ Hamming code

Four data bits (a_0, a_1, a_2, a_3)

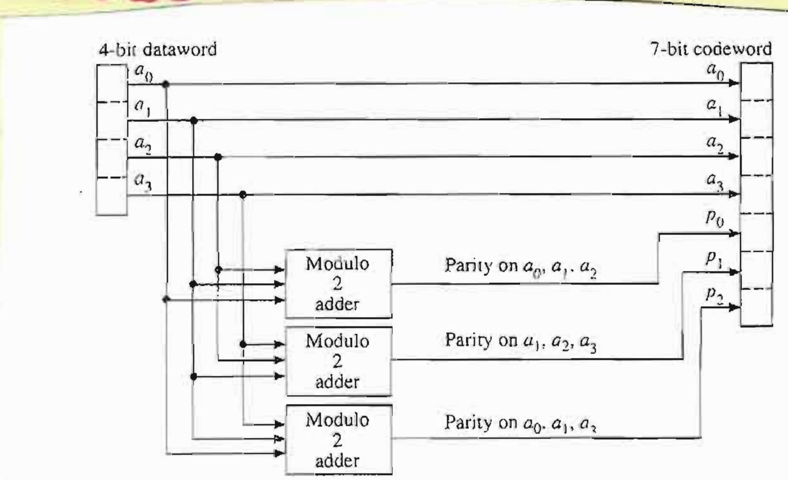
Codeword length is seven : $\underbrace{a_0 a_1 a_2 a_3}_{\text{original message}} \underbrace{p_0 p_1 p_2}_{\text{3 check bits}}$

Define check bits by:

$$p_0 = a_0 + a_1 + a_2$$
$$p_1 = a_1 + a_2 + a_3$$
$$p_2 = a_0 + a_1 + a_3$$

+ represents modulo-2 addition (bit by bit with no carries)

$$\begin{array}{r}0\\+0\\\hline 0\end{array} \quad \begin{array}{r}0\\+1\\\hline 1\end{array} \quad \begin{array}{r}1\\+0\\\hline 1\end{array} \quad \begin{array}{r}1\\+1\\\hline 0\end{array}$$



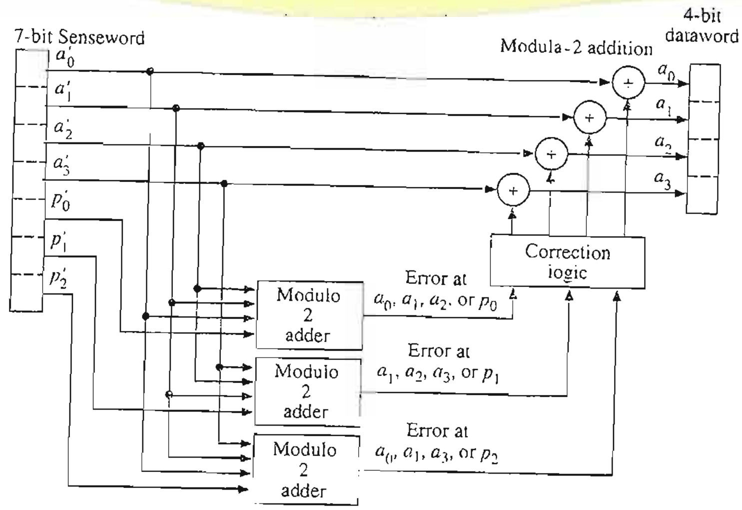
(a) Encoder

Table 1.1. *The* (7, 4) *Hamming code*

0	0	0	0	0	0	0
0	0	0	1	0	1	1
0	0	1	0	1	1	0
0	0	1	1	1	0	1
0	1	0	0	1	1	1
0	1	0	1	1	0	0
0	1	1	0	0	0	1
0	1	1	1	0	1	0
1	0	0	0	1	0	1
1	0	0	1	1	1	0
1	0	1	0	0	1	1
1	0	1	1	0	0	0
1	1	0	0	0	1	0
1	1	0	1	0	0	1
1	1	1	0	1	0	0
1	1	1	1	1	1	1

permuting bit positions ... one obtains an equivalent code

Decoder for (7,4) Hamming Code



(b) Decoder

The decoder receives a 7-bit senseword

$$v = (a_0', a_1', a_2', a_3', p_0', p_1', p_2')$$

and computes

$$S_0 = p_0' + a_0' + a_1' + a_2'$$
$$S_1 = p_1' + a_1' + a_2' + a_3'$$
$$S_2 = p_2' + a_0' + a_1' + a_3'$$

Syndrome - 3 bit pattern (S_0, S_1, S_2)

Reflects the error pattern

I

Syndrome Table for (7, 4) Hamming code

Syndrome	Error	
000	0000000	// no error has occurred
001	0000001	
010	0000010	// error in p_1
011	0001000	
100	0000100	
101	1000000	// error in a_0
110	0010000	
111	0100000	

A few examples:

message = $a_0\ a_1\ a_2\ a_3$
$\quad$ 0 0 1 1

$p_0 = a_0 + a_1 + a_2 = 0+0+1 = 1$
$p_1 = a_1 + a_2 + a_3 = 0+1+1 = 0$
$p_2 = a_0 + a_1 + a_3 = 0+0+1 = 1$

$\therefore$ code word = $a_0\ a_1\ a_2\ a_3\ p_0\ p_1\ p_2$
$\quad$ 0 0 1 1 1 0 1

Suppose, however, that $a_0'\ a_1'\ a_2'\ a_3'\ p_0'\ p_1'\ p_2'$
$\quad$ 0 0 0 1 1 0 1 is received

The decoder computes the syndrome

I

$a_0' \; a_1' \; a_2' \; a_3' \; p_0' \; p_1' \; p_2'$

$0 \; 0 \; \underline{0} \; 1 \; 1 \; 0 \; 1$ was received.

$$S_0 = p_0' + a_0' + a_1' + a_2' = 1+0+0+0 = 1$$
$$S_1 = p_1' + a_1' + a_2' + a_3' = 0+0+0+1 = 1$$
$$S_2 = p_2' + a_0' + a_1' + a_3' = 1+0+0+1 = 0$$

Syndrome = 110 which corresponds to an error pattern of 0010000

$$\begin{array}{rcl} 0\,0\,0\,1\,1\,0\,1 & = & \text{7-bit senseword} \\ +\;0\,0\,1\,0\,0\,0\,0 & = & \text{error pattern} \\ \hline 0\,0\,1\,1\,1\,0\,1 & & \text{most likely transmitted code word} \end{array}$$

0011 → most likely transmitted dataword.

What happens if two errors occur during transmission?

Let message = 1010

then codeword = $1010\,\underline{p_0}\,\underline{p_1}\,\underline{p_2} = 1010\underline{0}\,\underline{1}\,\underline{1}$

But suppose senseword 1111011 is received

Syndrome : $S_0 = 0+1+1+1 = 1$
$S_1 = 1+1+1+1 \quad 0$
$S_2 = 1+1+1+1 \quad 0$

error pattern is 0000100

What happens here?

Compact notation for (7,4) Hamming code.

Encoding process

codeword = Generator matrix: G · message

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ p_0 \\ p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \hdashline 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

Observe that upper 4×4 portion of the Generator matrix is the Identity matrix I_4.

Decoding process

Syndrome = Parity check matrix : H · sense word

$$\begin{bmatrix} s_0 \\ s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_0' \\ a_1' \\ a_2' \\ a_3' \\ p_0' \\ p_1' \\ p_2' \end{bmatrix}$$