ELEC 405
Error Control Coding and Sequences

Simple Encoding and Decoding

Equivalence of Linear Codes

Definition Two linear codes are called equivalent if one can be obtained from the other by the following operations:

(a) permutation of the positions of the code;

(b) multiplication of symbols in a fixed position by a non-zero element.

Theorem Two $k \times n$ matrices generate equivalent linear (n,k,d) codes if one matrix can be obtained from the other by a sequence of the following operations:

(a) permutation of the rows

(b) multiplication of a row by a non-zero element

(c) addition of one row to another

(d) permutation of the columns

(e) multiplication of a column by a non-zero element

Proof Operations (a) - (c) just replace one basis by another. The last two operations convert a generator matrix to one of an equivalent code.

Equivalence of Binary Linear Codes

Two binary linear codes are called equivalent if one can be obtained from the other by permuting the positions of the code

Two $k \times n$ binary matrices generate equivalent linear (n,k,d) codes if one matrix can be obtained from the other by a sequence of the following operations:

(a) permutation of the rows

(b) addition of one row to another

(c) permutation of the columns

Equivalent Codes

Distances between codewords are unchanged by the equivalence operations.

Consequently, equivalent linear codes have the same parameters (n,k,d) (and correct the same number of errors).

Therefore the form of the code can be chosen to best suit the application.

Systematic Codes

Theorem Let G be a generator matrix of an (n,k) code. Then by the previous operations, G can be transformed into the form

$$[\, I_k \mid A \,]$$

where I_k is the $k \times k$ identity matrix, and A is a $k \times (n - k)$ matrix.

Example

$$\begin{matrix} 1&1&1&1&1&1&1 \\ 1&0&0&0&1&0&1 \\ 1&1&0&0&0&1&0 \\ 1&1&1&0&0&0&1 \end{matrix} \rightarrow \begin{matrix} 1&1&1&1&1&1&1 \\ 0&1&1&1&0&1&0 \\ 0&0&1&1&1&0&1 \\ 0&0&0&1&1&1&0 \end{matrix} \rightarrow$$

$$\begin{matrix} 1&0&0&0&1&0&1 \\ 0&1&1&1&0&1&0 \\ 0&0&1&1&1&0&1 \\ 0&0&0&1&1&1&0 \end{matrix} \rightarrow \begin{matrix} 1&0&0&0&1&0&1 \\ 0&1&0&0&1&1&1 \\ 0&0&1&1&1&0&1 \\ 0&0&0&1&1&1&0 \end{matrix} \rightarrow ?$$

Systematic Codes

- For a systematic block code the data appears unaltered in the codeword – usually at the start
- The generator matrix has the structure

$$G = \overbrace{\begin{bmatrix} 1 & 0 & .. & 0 \\ 0 & 1 & .. & 0 \\ .. & .. & .. & .. \\ 0 & 0 & .. & 1 \end{bmatrix}}^{k} \overbrace{\begin{bmatrix} p_{0,0} & p_{0,1} & .. & p_{0,n-k-1} \\ p_{1,0} & p_{1,1} & .. & p_{1,n-k-1} \\ .. & .. & .. & .. \\ p_{k-1,0} & p_{k-1,1} & .. & p_{k-1,n-k-1} \end{bmatrix}}^{n-k} = \left[\mathrm{I} \mid \mathrm{P} \right]$$

- P is often referred to as the parity matrix

Encoding with a Systematic Code

- vector × matrix multiplication
- **Encoding** of a message $m = (m_0, \ldots, m_{k-1})$ with C

$$c = m \cdot \mathrm{G} = m\left[I_k \mid P\right] = (m \mid b)$$

Example Let C be a (7,4) code with generator matrix

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

Example (Cont.)

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

A message (m_0, m_1, m_2, m_3) is encoded as:

(0 0 0 0) is encoded as 0000000
(1 0 0 0) is encoded as 1000101
(1 1 1 0) is encoded as ?

Another Systematic Code

$$G = \begin{bmatrix} 1000 & 011 \\ 0100 & 101 \\ 0010 & 110 \\ 0001 & 111 \end{bmatrix} = [I_4 \quad P]$$

$$m = 0111 \quad \mathrm{c} = mG = 0111100 = m100$$

$$m = 1011 \quad \mathrm{c} = mG = 1011010 = m010$$

Parity Check Matrix

- Define a parity check matrix $\mathbf{H}$ (n-$k \times n$) such that

$$\mathbf{G}\mathbf{H}^T = \mathbf{0}$$

  $\mathbf{H}$ is a basis for the dual space

- If $\mathbf{G}$ is in systematic form

$$\mathbf{G} = [\mathbf{I}_k \ \mathbf{P}]$$

$$\mathbf{H} = [-\mathbf{P}^T \ \mathbf{I}_{n-k}] = [\mathbf{P}^T \ \mathbf{I}_{n-k}] \text{ (binary)}$$

$$\mathbf{G}\mathbf{H}^T = -\mathbf{P}+\mathbf{P} = \mathbf{0}$$

- Every codeword satisfies the parity check equations as given by the rows of $\mathbf{H}$

$$\mathbf{c}\mathbf{H}^T = \mathbf{m}\mathbf{G}\mathbf{H}^T = \mathbf{0}$$

Examples

- TRC (3,1,3)

$$\mathbf{G} = [1 \mid 1\ 1] \qquad \mathbf{H} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

(3,2,2) code

- SPC (8,7,2)

$$\mathbf{G} = \left[\begin{array}{c|c} & 1 \\ & 1 \\ & 1 \\ \mathbf{I}_7 & 1 \\ & 1 \\ & 1 \\ & 1 \end{array}\right] \qquad \mathbf{H} = [1\,1\,1\,1\,1\,1\,1 \mid 1]$$

(8,1,8) code

Dual Codes

If C is a linear (n,k) code, then the dual code $C^{\perp}$ is a linear $(n,n\text{-}k)$ code.

The dual code is just the dual subspace.

Example

$$C_6 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \text{ then } C_6^{\perp} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Dual Codes

For the (n,1) repetition code C with generator matrix

$$\mathbf{G} = [1,1, \ldots ,1]$$

the dual code $C^{\perp}$ is an (n,n-1) SPC code with generator matrix described by

$$\mathrm{H} = \begin{bmatrix} 1 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 0 & 1 & 0 & \ldots & 0 \\ & .. & & & & \\ 1 & 0 & 0 & 0 & \ldots & 1 \end{bmatrix}$$

The rows of **G** are orthogonal to the rows of **H**

Systematic Parity Check Matrices

If $\mathbf{G} = [\mathbf{I}_k \mid \mathbf{P}]$ is the standard form (systematic) generator matrix of a binary (*n,k*) code *C*, then a parity check matrix for *C* is $\mathbf{H} = [\mathbf{P}^T \mid \mathbf{I}_{n-k}]$

Example:

$$G = \left[\begin{array}{c|ccc} & 1 & 0 & 1 \\ & 1 & 1 & 1 \\ I_4 & 1 & 1 & 0 \\ & 0 & 1 & 1 \end{array}\right] \quad \Rightarrow \quad H = \left[\begin{array}{cccc|c} 1 & 1 & 1 & 0 & \\ 0 & 1 & 1 & 1 & I_3 \\ 1 & 1 & 0 & 1 & \end{array}\right]$$

Dual of the (5,2,3) Code

- Consider two equivalent (5,2,3) codes - nonsystematic and systematic

$$\mathbf{G} = \begin{bmatrix} 00111 \\ 11100 \end{bmatrix} \rightarrow \mathbf{G}' = \left[\begin{array}{c:cc} 10 & 110 \\ 01 & 101 \end{array}\right]$$

$$\mathbf{H} = \begin{bmatrix} 01101 \\ 00011 \\ 11000 \end{bmatrix} \quad \mathbf{H}' = \left[\begin{array}{c:cc} 11 & 100 \\ 10 & 010 \\ 01 & 001 \end{array}\right]$$

- **H** and **H'** both generate (5,3,2) codes

Decoding Linear Codes

- One possibility is a ROM look-up table
- In this case the received word **r** is used as an address
- Example – Even single parity check code

Address	Entry
00000000	0
00000001	1
00000010	1
00000011	0
……	.

- The output is an error flag, i.e., 0 – codeword valid
- If there is no error, the data is the first *k* codeword bits
- For an error correcting code, the ROM can also store the data after correction

Standard Array

- The standard array is formed by initially choosing $\mathbf{e}_i$ to be
  - All zero pattern
  - All 1 bit error patterns
  - All 2 bit error patterns

    ……
- Ensure that each new error pattern is not already in the array
- All correctable error patterns will appear as coset leaders

Standard Array

c_1 (all zero)	$\mathbf{c_2}$	……	c_{M-1}	c_M
e_1	c_2+e_1	……	$c_{M-1}+e_1$	c_M+e_1
e_2	c_2+e_2	……	$c_{M-1}+e_2$	c_M+e_2
$\mathbf{e_3}$	$\mathbf{c_2+e_3}$	……	$c_{M-1}+e_3$	c_M+e_3
….	……	……	……	…..
$e_{2^{n-k}-1}$	$c_2+ e_{2^{n-k}-1}$	……	$c_{M-1}+ e_{2^{n-k}-1}$	$c_M+ e_{2^{n-k}-1}$

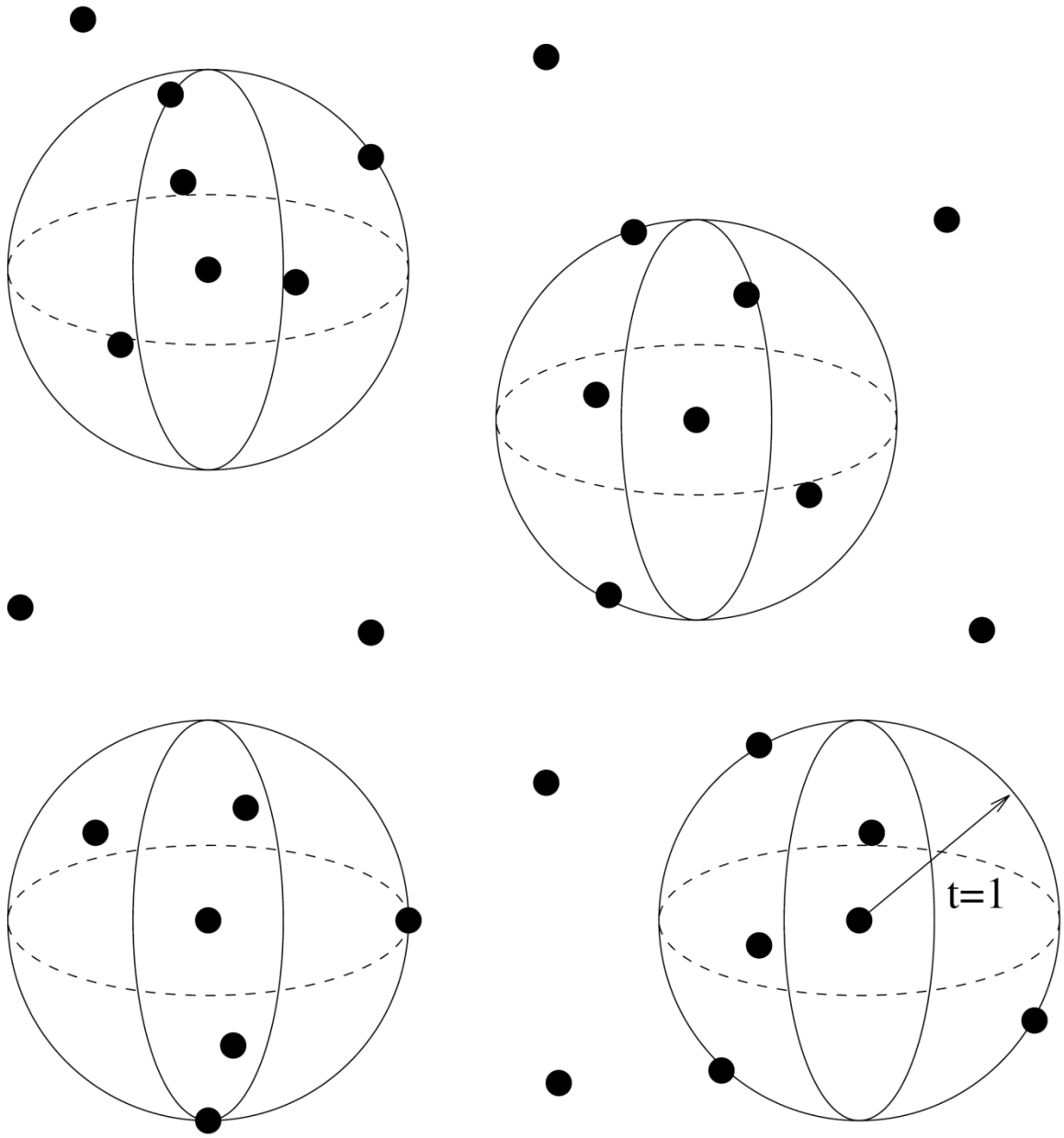
The array has 2^k columns (equal to the number of codewords) and 2^{n-k} rows (the number of correctable error patterns)

Standard Array Decoding

- Assume that the received word is $\mathbf{r} = \mathbf{c}_2 + \mathbf{e}_3$ (shown in bold in the standard array)
- The most likely codeword is the one at the top of the column containing $\mathbf{c}_2 + \mathbf{e}_3$
- The corresponding error pattern is the coset leader at the start of the row containing $\mathbf{c}_2 + \mathbf{e}_3$
- Can be implemented using a look-up table (ROM) which maps all words in the array to the most likely codeword (the one at the top of the column containing the received word).

Standard Array for the (5,2,3) Code

00000	10110	01101	11011
10000	00110	11101	01011
01000	11110	00101	10011
00100	10010	01001	11111
00010	10100	01111	11001
00001	10111	01100	11010
00011	10101	01110	11000
10001	00111	11100	01010


t=1

Syndromes

- For a received word **r**, we need a simpler method to determine the codeword estimate
- To do this we use the syndrome, **s** of a received word **r**

$$\mathbf{s} = \mathbf{r}\mathbf{H}^T$$

- If *c* was corrupted by an error vector, **e**, then

$$\mathbf{r} = \mathbf{c} + \mathbf{e}$$

and

$$\mathbf{s} = \mathbf{r}\mathbf{H}^T = (\mathbf{c} + \mathbf{e})\mathbf{H}^T = \mathbf{c}\mathbf{H}^T + \mathbf{e}\mathbf{H}^T$$

$$\mathbf{s} = \mathbf{0} + \mathbf{e}\mathbf{H}^T$$

- The syndrome depends only on the error vector **e**

Syndromes

- We can add the same error pattern to different codewords and get the same syndrome.

- There are $2^{(n - k)}$ syndromes but 2^n received words
  - (5,2,3) code has 8 syndromes and 32 received words
  - (7,4,3) code has 8 syndromes and 128 received words

- Only need to determine which error pattern corresponds to the syndrome

Syndromes for the (5,2,3) Code

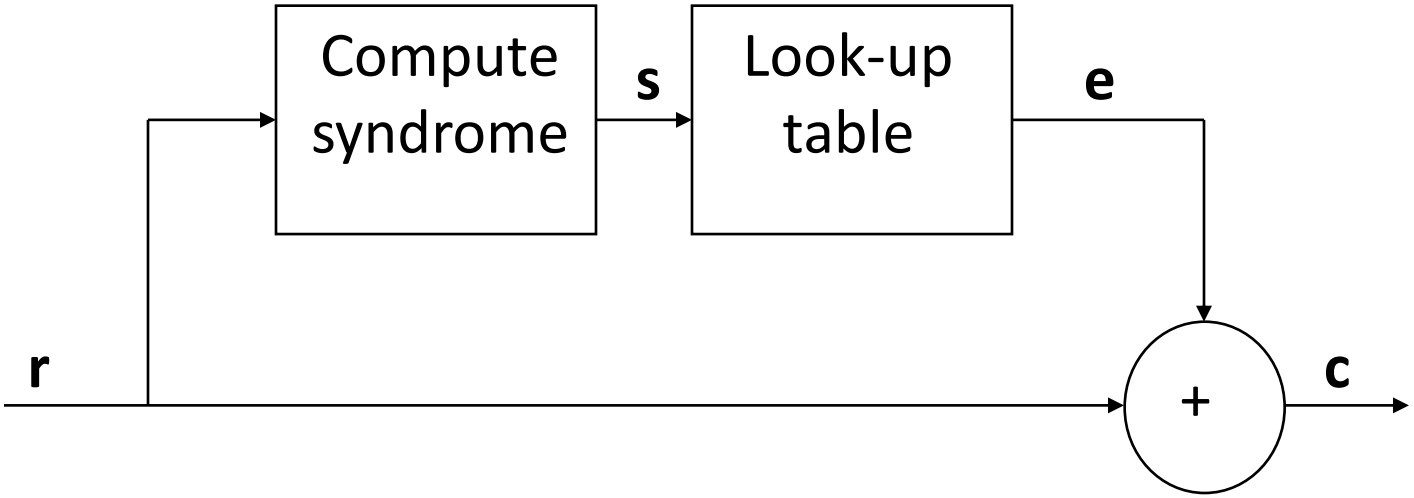
$$G = \left[\begin{array}{cc|ccc} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{array}\right]$$

$$H = \left[\begin{array}{cc|ccc} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{array}\right]$$

$$\begin{aligned} s_0 &= 00000H^T = 000 \\ s_1 &= 10000H^T = 110 \\ s_2 &= 01000H^T = 101 \\ s_3 &= 00100H^T = 100 \\ s_4 &= 00010H^T = 010 \\ s_5 &= 00001H^T = 001 \\ s_6 &= 00011H^T = 011 \\ s_7 &= 10001H^T = 111 \end{aligned}$$

Syndrome Decoding

- This block diagram shows the syndrome decoder implementation



Syndrome Decoding

- For systematic linear block codes

  $\mathbf{G} = [\, \mathbf{I} \mid \mathbf{P} \,]$ so $\mathbf{H} = [\, -\mathbf{P}^T \mid \mathbf{I} \,]$

- Example (7,4,3) code

$$G = [I \mid P] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$H = [-P^T \mid I] = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Syndrome Decoding – Example 1

- For a received word **r** = (1101001)

$$\mathbf{s} = \mathbf{r}\mathbf{H}^{\mathrm{T}} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$$

- So **r** is a valid codeword

Syndrome Decoding – Example 2

- The same codeword, this time with an error in the first bit position **r** = (1101000)

$$\mathbf{s} = \mathbf{r}\mathbf{H}^{\mathrm{T}} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

- Syndrome 001 indicates an error in bit 7 of the received word