Binary Error Correcting Codes

1 Basic concepts of Error correcting Codes

In communication system, we represent an information as a sequence of 0 an 1 (binary form). For a convenience, let $B = \{0, 1\}$. Then we define $B^2, B^3, \ldots, B^n$ as follows :

$$\begin{aligned} B^2 &= \{00, 01, 10, 11\}, \\ B^3 &= \{000, 001, 010, 100, 011, 101, 110, 11\}, \\ &\vdots \\ B^n &= \{b_1 b_2 \ldots b_n | b_i \in B\} \end{aligned}$$

A symbol $b_1 b_2 \ldots b_n \in B^n$ is called a *word*. We always denote $\mathbf{0}$ and $\mathbf{1}$ for $00 \ldots 0$ and $11 \ldots 1$, respectively.
We define binary operations $+, \cdot : B \times B \rightarrow B$ as follows :

+	0	1
0	0	1
1	1	0

·	0	1
0	0	0
1	0	1

Clearly, $(B, +)$ is an abelian group.

Exercise 1.1. *Let $b_1 b_2 \ldots b_n, c_1 c_2 \ldots c_n \in B^n$ and for each $i = 1, 2, \ldots, n$, let $d_i = b_i + c_i$ as above table. Define a binary operation $+ : B^n \times B^n \rightarrow B^n$ by*

$$(b_1 b_2 \ldots b_n, c_1 c_2 \ldots c_n) \mapsto d_1 d_2 \ldots d_n.$$

i) verify that $(B^n, +)$ *is an abelian group,*

ii) for each $b_1b_2 \ldots b_n \in B^n$*, determine its inverse.*

The following diagram provides a rough idea of general information transmitted system.

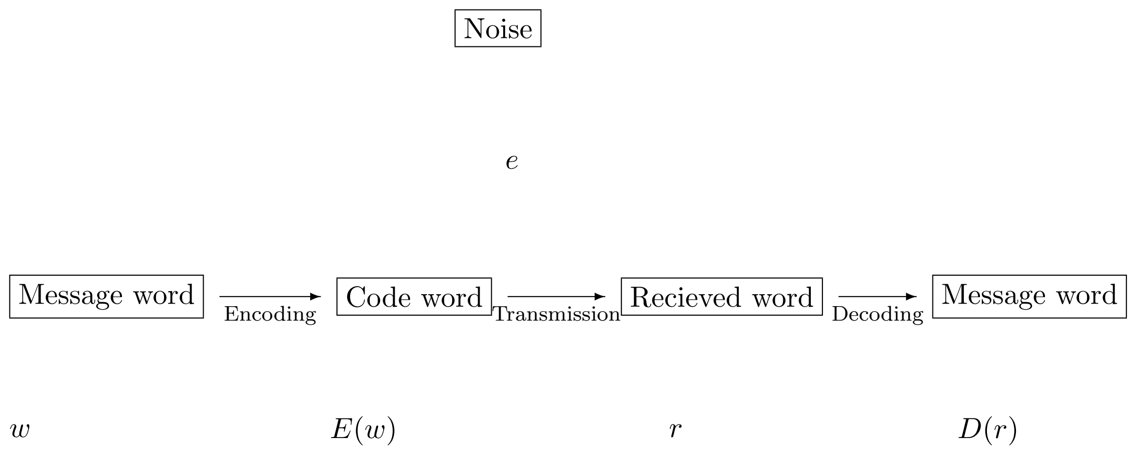

Fig.1 : The communication channel

From above figure, we give concepts of a binary (n, m) code as follows:

Definition 1.1. Let $k, n \in \mathbb{N}$ be such that $m < n$. A *binary* (n, m) *code* (or *code*) compose of :

1. an injective function $E : B^m \to B^n$, called an *encoding function*,
2. a function $D : B^n \to B^m$ such that $D(E(w)) = w$ for all $w \in B^m$, called a *decoding function.*

We call a set $M \subseteq B^m$ a *set of massage*, $w \in M$ a *message word*, $\mathcal{C} := E(M)$ a *code*, $c \in \mathcal{C}$ a *code word*, $r \in Dom(D)$ a *received word* .

In general, $M \neq B^m$. WLOG, we assume for a convenience that $M = B^m$. Then a code $\mathcal{C} := E(M) = E(B^m)$ and $|\mathcal{C}| = 2^m$.

Definition 1.2. Let $\mathcal{C} \subseteq B^n$ be a code and $c \in \mathcal{C}$. If a word r is received (from c) and $e \in B^n$ is such that $r = c + e$, we call e an *error* (or *error pattern*).

Example 1.1 (Even parity-check code). *We define*

$$E : B^m \to B^{m+1} \text{ by } b_1b_2\dots b_m \mapsto b_1b_2\dots b_m b_{m+1}$$

where

$$b_{m+1} = \begin{cases} 0 \text{ if the number of } 1s' \text{ in } b_1b_2\dots b_m \text{ is even} \\ 1 \text{ if the number of } 1s' \text{ in } b_1b_2\dots b_m \text{ is odd} \end{cases}$$

and

$$D : B^{m+1} \to B^m$$

by

$$b_1b_2\dots b_m b_{m+1} \mapsto \begin{cases} b_1b_2\dots b_m & \text{if the number of } 1s' \text{ in } b_1b_2\dots b_m \text{ is even} \\ 00\dots 0 & \text{if the number of } 1s' \text{ in } b_1b_2\dots b_m \text{ is odd} \end{cases}$$

Then even parity-check code is an $(m+1, m)$ code.
For example, B^3 is encoded as follow :

message word	000	001	010	100	011	101	110	111
code word	0000	0011			0110			

The following received words are decoded as in the table :

received word	1110	0101	0110	0001	1010	1101
message word	000				101	

Example 1.2 (Triple-repetition code). *Triple-repetition code is* $(3m, m)$ *code such that an encoding function*

$$E : B^m \to B^{3m}$$

is defined by

$$b_1 b_2 \dots b_m \mapsto b_1 b_2 \dots b_m b_1 b_2 \dots b_m b_1 b_2 \dots b_m$$

and a decoding function

$$D : B^{3m} \to B^m$$

is defined by

$$x_1 x_2 \dots x_m y_1 y_2 \dots y_m z_1 z_2 \dots z_m \mapsto b_1 b_2 \dots b_m$$

where

$$b_i \mapsto \begin{cases} 0 & \text{if 0 occurs in } x_i y_i z_i \text{ at least twice} \\ 1 & \text{if 1 occurs in } x_i y_i z_i \text{ at least twice} \end{cases}$$

For example, B^3 is encoded as follow :

message	000	001	010	100	011	101	110	111
code word	000 000 000		010 010 010					

The following received words are decoded as in the table :

received word	101 101 101	010 111 110	011 101 110	001 101 001	111 000, 101
message word	101				

Moreover, $n-$repetition code is defined similarly.

Nearest Neighbor Decoding : For a code $\mathcal{C}$, if a word r is received, it is decoded as the code word in $\mathcal{C}$ closest to it.
Complete Nearest Neighbor Decoding : If more than one candidate appears, choose arbitrarily.
Incomplete Nearest Neighbor Decoding : If more than one candidate appears, request a retransmission.

To measure a distance between any two code words, we introduce the Hamming distance as follow :

Definition 1.3. Let $u = u_1u_2 \dots u_n$, $v = v_1v_2 \dots v_n \in B^n$. The *distance* $d(u, v)$ of u and v is defined by

$$d(u, v) = |\{i \in \{1, 2, \dots, n\}|u_i \neq u_i\}|.$$

The *weight* $w(u)$ of u is defined by

$$w(u) = |\{i \in \{1, 2, \dots, n\}|u_i \neq 0\}|$$

The distance and weight defined above are called the *Hamming-distance* and *Hamming-weight* , respectively.

Lemma 1.1. *Let* $u, v \in B^n$*. Then* $w(u) = d(u, \boldsymbol{0})$ *and* $d(u, v) = w(u + v)$*.*

Lemma 1.2. *Let* $u, v, w \in B^n$*. Then*

i) $d(u, v) \geq 0$*,*

ii) $d(u, v) = 0$ *iff* $u = v$*,*

iii) $d(u, v) = d(v, u)$*,*

iv) $d(u, v) \leq d(u, w) + d(w, v)$*,*

and hence (B^n, d) *is a metric space.*

Example 1.3. *Let* $\mathcal{C} = \{0000000, 1001100, 1101101, 0110011\}$ *be a* $(7, 2)$ *code.*

The following table displays Hamming weight of each code word in $\mathcal{C}$ *:*

code word v	Hamming weight *w(v)*
0000000	
1001100	3
1101101	
0110011	

The following table displays H-distance between any two code words in $\mathcal{C}$:

d	0000000	1001100	1101101	0110011
0000000	0	3		
1001100				
1101101				
0110011			5	

Assume that complete nearest neighbor decoding is used. We introduce two methods to decode received words. Let r be a received word.

1. *Find the closest code word $v \in \mathcal{C}$ such that $d(r, v) \leq d(r, u)$ for all $u \in \mathcal{C}$:*

2. *Since $d(r, b) = w(r + b)$ for all $b \in B^n$, r is decoded to $v \in \mathcal{C}$ such that $w(r + v) \leq w(r + u)$ for all $u \in \mathcal{C}$*

Assume that $0001001, 1010100, 1001001, 0100101, 1110100, 1111111$ *are received words. We decode them as follows :*

By 1^{st} *method,*

d	0000000	1001100	1101101	0110011	decode to
0001001	2				
1010100	3	<u>2</u>	4	5	1001100
1001001					
0100101					
1110100					
1111111					

By 2^{nd} *method,*

+	0000000	1001100	1101101	0110011	decode to
0001001	<u>0001001</u>	1000101	1100100	0111010	0000000
1010100	1010100	<u>0011000</u>	0111001	1100111	1001100
1001001			0101101	1110011	
0100101					
1110100					
1111111					

Example 1.4. *Let*
$\mathcal{C} = \{0111000, 0010010, 1101101, 1001000, 1100010, 0011101, 0110111, 1000111\}$
be a $(7,4)$ *code. Assume that* 0001001, 1010100, 1001001, 0100101, 1110100, 1111111 *are received words. We decode them by* 2^{nd} *method,*

+	0111000	0010010	1101101	1001000	1100010	0011101	0110111	1000111	decode to
0001001									
1010100									
1001001									
0100101									
1110100									
1111111									

Definition 1.4. Let $\mathcal{C}$ be a code such that $|\mathcal{C}| \neq 1$. The *minimum distance* $d(\mathcal{C})$ of $\mathcal{C}$ is

$$d(\mathcal{C}) = \min\{d(u,v) | u, v \in \mathcal{C}, u \neq v\}.$$

The *minimum weight* $w(\mathcal{C})$ of $\mathcal{C}$ is

$$w(\mathcal{C}) = \min\{w(u) | u \in \mathcal{C} \backslash \{\mathbf{0}\}\}.$$

The minimum distance of a code tell me about the correction (and detection) capability of its.

Theorem 1.3. *Let $\mathcal{C} \in B^n$ be a code. Assume that nearest neighbor decoding is used. Then*

1) If $t+1 \leq d$, then $\mathcal{C}$ can detect $t-$errors.

2) If $2l+1 \leq d$, then $\mathcal{C}$ can correct $l-$errors.

Example 1.5. *Refer to codes in above examples.*

1. *Even parity check code in Example 1.1 has the minimum distance 2 and hence it can detect at most $1-$error but cannot correct any error. (Verify !)*
2. *Triple-repetition code in Example 1.2 has the minimum distance $\boxed{3}$ and hence it can detect at most $\boxed{}$-error(s) and can correct at most $\boxed{}$-error(s). (Verify !)*
3. *A code $\mathcal{C}$ in Example 1.3 has the minimum distance $\boxed{}$ and hence it can detect at most $\boxed{}$-error(s) and can correct at most $\boxed{}$-error(s).*
4. *A code $\mathcal{C}$ in Example 1.4 has the minimum distance $\boxed{}$ and hence it can detect at most $\boxed{}$-error(s) and can correct at most $\boxed{}$-error(s).*

Example 1.6. *Let* $\mathcal{C} = \{00000000, 11101011, 01011110, 10110101\}$ *be a* $(8, 2)$ *code. Distance between any two code words display on the table :*

d	00000000	11101011	01011110	10110101
00000000	0	6	5	5
11101011	6	0	5	5
01011110	5	5	0	6
10110101	5	5	6	0

Then $\mathcal{C}$ *has the minimum distance* 5*. This means that can correct at most* $2-$*errors.*
Assume complete nearest neighbor decoding is used. If words 11111111, 00001011 *and* 11110000 *are received, we can decode as follow :*

+	00000000	11101011	01011110	10110101	decode to	describtion
11111111	11111111	<u>00010100</u>	10100001	01001010	11101011	can correct 2−errors
00001011	<u>00001011</u>	<u>11100000</u>	01010101	10111110	choose arbitrarily	cannot correct some 3−errors
11110000	11110000	00011011	10101110	<u>01000101</u>	10110101	can correct some 3−errors

When size of code is large, the minimum distance of code is hard to compute. Next, we introduce you a more efficiency code which is called a linear code (or group code).

2 Linear Codes (group codes)

Recall that $(B^n, +)$ is an abelian group.

Definition 2.1. A (n, k) code $\mathcal{C} \subseteq B^n$ is called a *linear code* (or *group code*) if for all $u, v \in \mathcal{C}$, $u + v \in \mathcal{C}$.

Exercise 2.1. *Let $\mathcal{C} \subseteq B^n$ be a code. Verify that "$\mathcal{C}$ is a linear code if and only if $\mathcal{C}$ is a subgroup of B^n ".*

Since $\mathcal{C}$ is a subgroup of B^n, by Lagrange's Theorem $|\mathcal{C}|\,|\,|B^n| = 2^n$ and hence $|\mathcal{C}| = 2^k$ for some $k \in \{0, 1, 2, \ldots, n\}$. This means that $\mathcal{C}$ contain 2^k words of length n.

Definition 2.2. We call a linear code $\mathcal{C} \subseteq B^n$ with $|\mathcal{C}| = 2^k$ an $[n, k]$ code . If an $[n, k]$ code $\mathcal{C}$ has the minimum distance $\mathcal{C}$, we call $\mathcal{C}$ an $[n, k, d]$ code.

Example 2.1. *Refer to codes in above examples.*

1. *Even parity check code in Example 1.1 is a linear code with the minimum distance* 2*. Hence it is a* $[m + 1, m, 1]$ *code. (Verify !)*
2. *Triple-repetition code in Example 1.2 is a linear code with the minimum distance* 3*. Hence it is a* $[3m, m, 3]$ *code. (Verify !)*
3. *A code $\mathcal{C}$ in Example 1.6 is a* $[8, 2, 5]$ *code.(Verify !)*

Theorem 2.1. *Let $\mathcal{C} \subseteq B^n$ be a linear code. Then $d(\mathcal{C}) = w(\mathcal{C})$.*

Example 2.2. *Consider the code*
$\mathcal{C} = \{000000, 001110, 010101, 011011, 100011, 101101, 110110, 111000\}$. *Then $\mathcal{C}$ is a linear code (verify!) and hence $\mathcal{C}$ has the minimum distance $d(\mathcal{C}) = w(\mathcal{C}) = 3$, i.e., $\mathcal{C}$ is a* $[6, 3, 3]$ *code.*

Example 2.3. *Consider the code* $\mathcal{C} = \{111111, 100110, 010001, 011010\}$. *Then* $\mathcal{C}$ *has the minimum distance* $d(\mathcal{C}) = 3$ *is not equal to* $w(\mathcal{C}) = 2$. *Why?*

For any code, we can decode by methods which described in Example 1.3. Now, If $\mathcal{C}$ is a linear code, we have more efficiency methods.

2.1 Cosets and Coset Decoding

Since an $[n, k]$ code $\mathcal{C}$ is a subgroup of B^n, for $u \in B^n$, $u+\mathcal{C} = \{u+v|v \in \mathcal{C}\}$ is called a *coset of* $\mathcal{C}$ *generated by* u. Clearly, the number of all (distinct) coset of $\mathcal{C}$ is $[B^n : \mathcal{C}] = \dfrac{2^n}{2^k} = 2^{n-k}$.

Definition 2.3. For a coset $u + \mathcal{C}$, we call $v \in u + \mathcal{C}$ a *coset leader* if $w(v) \leq w(u+\mathcal{C})$.

Note that a coset leader may not unique.

Example 2.4. *Consider a code* $\mathcal{C} = \{0000, 0110, 1011, 1101\}$. *Then* $\mathcal{C}$ *is a linear* $[4, 2, 2]$ *code. Then we obtain cosets and coset leaders (underline words) :*

$\mathcal{C} + 0000$	$\mathcal{C} + 0100$	$\mathcal{C} + 1000$	$\mathcal{C} + 0001$
$\underline{0000}$	$\underline{0100}$	$\underline{1000}$	$\underline{0001}$
0110	$\underline{0010}$	1110	0111
1011	1111	0011	1010
1101	1001	0101	1100

The above table is called the standard decoding array *(or* standard array*).*

Coset Decoding: Let $\mathcal{C}$ be an $[n, k]$ code. If a word $r \in B^n$ is received and v is the coset leader for $r + \mathcal{C}$, then decode r as $r + v$.

Theorem 2.2. *Coset decoding is nearest neighbor decoding.*

Proof. Let $\mathcal{C}$ be an $[n, k]$ code , $u \in B^n$ and v be a coset leader for $u + \mathcal{C}$. Since $v \in u + \mathcal{C}$, $u + \mathcal{C} = v + \mathcal{C}$ and hence $v := u + v \in \mathcal{C}$. Let $x \in \mathcal{C}$. Then $u + x \in u + \mathcal{C} = v + \mathcal{C}$, i.e., $w(v) \leq w(u + x)$. Thus

$$d(v, u) = w(u + v) = w(v) \leq w(u + x) = d(u, v).$$

□

Example 2.5. *Consider the standard array*

$\mathcal{C} + 0000$	$\mathcal{C} + 0100$	$\mathcal{C} + 1000$	$\mathcal{C} + 0001$
<u>0000</u>	<u>0100</u>	<u>1000</u>	<u>0001</u>
0110	<u>0010</u>	1110	0111
1011	1111	0011	1010
1101	1001	0101	1100

.

Assume that coset decoding is used. If words 0101, 1010, 1111, 1011, 0111 *are received, then we decode them as* $r + v$ *where* r *is a received word and* v *is a coset leader :*

received word (r)	decode to $(r + e)$
0101	$0101 + 1000 = 1101$
1010	
1111	
1011	
0111	

.

Example 2.6. *Construct the standard array for the linear* $[6, 3, 3]$ *code*

$$\mathcal{C} = \{000000, 001110, 010101, 011011, 100011, 101101, 110110, 111000\}.$$

$\mathcal{C} + 000000$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$
000000							
001110							
010101							
011011							
100011							
101101							
110110							
111000							

.

Assume that coset decoding is used. Decode followings received words :

received word (r)	decode to $(r + v)$
010101	
101011	
111111	
101100	
011110	
000111	
111110	

.

Describe about correction capability ?

2.2 Generator Matrix, Parity-check Matrix and Decoding

For a convenience, we consider a word $w = w_1w_2\ldots w_k \in B^k$ as a matrix $w = [\ w_1 \ \ w_2 \ \ \cdots \ \ w_k\ ]$. Let G be a binary $k \times n$ matrix such that $k < n$. Then $wG = [\ w_1 \ \ w_2 \ \ \cdots \ \ w_k\ ] \in B^n$ for all $w \in B^k$.

Definition 2.4. Let G be a binary $k \times n$ matrix such that $k < n$ and the first k columns is an identity matrix I_k. Define $E : B^k \to B^n$ by $E(w) = wG$. Then $\mathcal{C} := \{wG|w \in B^k\}$ is called a *code generated by* G and G is called the *(standard) generator matrix* for $\mathcal{C}$.

From the above definition, we write $G = [I_k \ \ A]$ for some $(k \times (n-k)$ matrix A. Then for each message word $u \in B^k$, $uG = [uI_k \ uA] = [u \ uA]$ which is easy to retrieve.

Exercise 2.2. *Verify the followings :*

i) E is an encoding function (i.e., E is injective).

ii) $\mathcal{C}$ is a linear code.

Definition 2.5. A binary $(n-k) \times n$ matrix H with $k < n$ is called the *(standard) parity-check matrix for a linear* $[n, k]$ *code* $\mathcal{C}$ if the last $n-k$ columns is an identity matrix I_{n-k} and $Hv^t = [\mathbf{0}]$ for all $v \in \mathcal{C}$.

Lemma 2.3. *If G and H are generator matrix and parity-check matrix for a linear code $\mathcal{C}$,respectively, then $HG^t = [\boldsymbol{0}]$*

Theorem 2.4. *If $G = [I_k \;\; A]$ is a generator matrix for a linear $[n, k]$ code $\mathcal{C}$, then $H = [A^t \;\; I_{n-k}]$ is a parity check matrix for $\mathcal{C}$.*
Conversely, if $H = [B \;\; I_{n-k}]$ is a parity check for a linear $[n, k]$ code $\mathcal{C}$, then $G = [I_k \;\; B^t]$ is a generator matrix for $\mathcal{C}$.

Example 2.7. *Even parity check code in Example 1.1 is a linear code with the generator matrix*

$$G = \left[\begin{array}{c|c} & 1 \\ I_m & \vdots \\ & 1 \end{array}\right].$$

Determine the parity-check matrix for even parity check code?
Triple-repetition code in Example 1.2 is a linear code with the generator matrix

$$G = \left[\begin{array}{c|c|c} I_m & I_m & I_m \end{array}\right].$$

Determine the parity-check matrix for triple-repetition code code?

Example 2.8. *Let*

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

Then

1. *The linear code*

$$\begin{aligned} \mathcal{C} &:= \{wG | w \in B^3\} \\ &= \{ \qquad\qquad\qquad\qquad\qquad\qquad \}. \end{aligned}$$

2. *The parity-check matrix*

$$H = \begin{bmatrix} \\ \quad\quad\quad\quad\quad\quad\quad\quad \\ \\ \end{bmatrix}$$

3. *All cosets and coset leaders*

$\mathcal{C} + 000000$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$

.

Example 2.9. *Let*

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

Then

1. *The linear code*

$$\begin{aligned} C &:= \{wG | w \in B^4\} \\ &= \{ \qquad \qquad \}. \end{aligned}$$

2. *The parity-check matrix*

$$H = \begin{bmatrix} \qquad \\ \qquad \\ \qquad \end{bmatrix}$$

3. All cosets and coset leaders

$\mathcal{C}+000000$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$	$\mathcal{C}+$

.

Definition 2.6. Let H be the parity-check matrix for a linear $[n, k]$ code $\mathcal{C}$. For each $v \in B^n$, the syndrome $S(v)$ of v is defined by $S(v) = Hv^t$

Theorem 2.5. *Let H be the parity-check matrix for a linear $[n, k]$ code $\mathcal{C}$ and $u, v \in B^n$. Then* *i)* $S(u+v) = S(u) + S(v)$,

ii) $S(v) = [\boldsymbol{0}]$ *if and only if* $v \in \mathcal{C}$,

iii) $S(u) = S(v)$ *if and only if* u *and* v *are in the same coset.*

Definition 2.7. A table which matches each coset leader e with its syndrome is called a *syndrome look-up table.*

Syndrome Decoding Let H be the parity-check matrix for a linear $[n, k]$ code $\mathcal{C}$. If $r \in B^n$ is received, compute $S(r)$ and find v (in a syndrome look-up table) such that $S(r) = S(v)$. Decode r as $r + v$.

Example 2.10. *Construct a syndrome look-up table for a* $[6, 3]$ *code in Example 2.8.*

coset leader v	syndrome $S(v)$

Assume that syndrome decoding is used. Decode following received words :

received word (r)	$S(r)$	decode to $(r+v)$ s.t. $S(r)=S(v)$
010101		
101011		
111111		
101100		
011110		
000111		
111110		

.

Exercise 2.3. *Construct a syndrome look-up table for a* $[7,4]$ *code in Example 2.9. Assume that syndrome decoding is used. Then decode following received words :* $0001001, 1010100, 1001001, 0100101, 1110100, 1111111.$

Parity-heck Matrix Decoding Let H be the parity-check matrix for a linear $[n,k]$ code $\mathcal{C}$. If $r \in B^n$ is received, compute $S(r) = Hr^t$.

1. If $S(r) = [\mathbf{0}]$, then $r \in \mathcal{C}$ and hence decode r as r.

2. If $S(r) \neq [\mathbf{0}]$ and $S(r)$ is column i of H, decode by changing its i^{th} bit.

3. If $S(r) \neq [\mathbf{0}]$ and $S(r)$ is not a column of H, request a retransmission.

Exercise 2.4. *For a* $[7,4]$ *code in Example 2.9. Assume that parity-check matrix decoding is used. Then decode followings received words :*

$$0001001, 1010100, 1001001, 0100101, 1110100, 1111111$$

.

References

[1] F.J. MacWilliams and N.J.A. Sloan, *The Theory of Error-Correcting Codes.* New York:Elsevier/North Halland, 1977.

[2] San Ling and Chaoping Xing, *Coding Theory : A First Course.* Cambridge University Press, 2004.

[3] Vera Pless, *Introduction to the Theory of Error-Correcting Codes.*, John Wiley and Son, 1990.

[4] J.H. Van Lint, *Graduate Texts in Matematics : Introduction to Coding Theory.* Spriger-Verlag,1982.

[5] D.G. Hoffman et al, *Algebraic Coding Theory.*Winnipeg/Canada, 1987.

[6] W.K.Nicholson, *Introduction to Abstract Algebra Algebra.*John Wiley & Sons, 1999.