E2 205 Error-Control Coding
Lecture 10

Scribe - Johns U K

September 16, 2019

1 Maximum Likelihood Decoding

Setting
Consider transmitting of a binary codeword over a binary input channel as in Fig 1. Also Consider an AWGN channel as in Fig 2. Each w_i is *iid*. The coding scheme employed is Binary Antipodal.

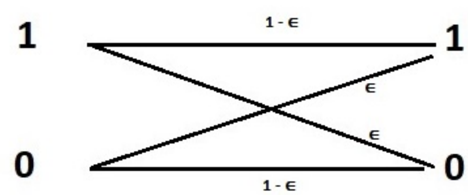

Figure 1: Binary Symmetric Channel

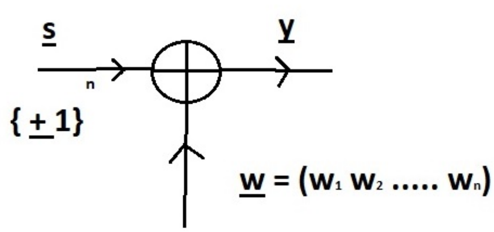

Figure 2: AWGN Channel

Let $\mathcal{C}$ be a binary code and let $M = |\mathcal{C}|$. Let y^n be the space of received vectors. The role of decoder is to partition y^n into M regions, one for each code word as in Fig 3 below.

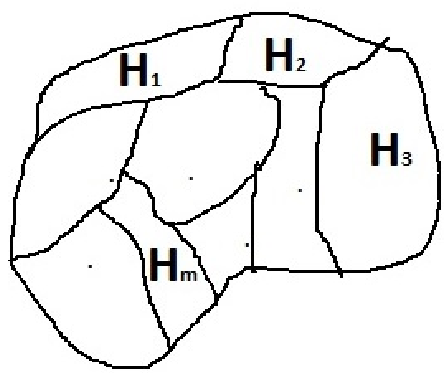

Figure 3: Received Vector Space Partitioning

If $\vec{y} \in H_i$, the decoder declares the estimated (decoded) codeword $\hat{c} = \vec{c_i}$ where

$$\mathcal{C} = \{\vec{c_i} : 1 \leq i \leq M\}. \tag{1}$$

Let ε denote the codeword error event i.e. event of decoding to an incorrect codeword. Then, ε^c denotes the correct decoding event.

$$P(\varepsilon^c) = \sum_{i=1}^{n} P(\vec{c_i}) P(\varepsilon^c | \vec{c_i})$$

$$
\begin{aligned}
P(\varepsilon^c) &= \sum_{i=1}^{n} P(\vec{c_i})P(\ \vec{y} \in H_i\ |\vec{c_i}) \\
&= \sum_{i=1}^{n} P(\vec{c_i}) \int_{H_i} P(\vec{y}|\vec{c_i})dy \\
&= \sum_{i=1}^{n} P(\vec{c_i}) \int_{y^n} P(\vec{y}|\vec{c_i})\mathbb{1}_{\mathbb{i}}(\vec{y})dy,\ where\ \mathbb{1}_{\mathbb{i}}(\vec{y}) = \{1,\ if\ \vec{y} \in H_i\ and\ 0\ elsewhere\} \\
&= \int_{y^n} \left[\sum_{i=1}^{n} P(\vec{c_i})P(\vec{y}|\vec{c_i})\mathbb{1}_{\mathbb{i}}(\vec{y})dy\right]
\end{aligned}
$$

Therefore, P($\vec{c_i}$) P($\vec{y}|\vec{c_i}$) is the contribution(for a given $\vec{y}$) to the problem of correct decoding if $\vec{y} \in H_i$. It follows that, to minimise the P(ε) given by $\vec{y}$ $\in y^n$, we assign $\vec{y} \in H_i$ such that,
P($\vec{c_i}$) P($\vec{y}|\vec{c_i}$) $\geq$ P($\vec{c_j}$) P($\vec{y}|\vec{c_j}$), $i \neq j$.

This is known as Minimum Probability of Error Decoding (MPE).

If P($\vec{c_i}$) = P($\vec{c_j}$) = $\frac{1}{M}$, i.e. if the codewords are equally likely, the above equation reduces to P($\vec{y}|\vec{c_i}$) $\geq$ P($\vec{y}|\vec{c_j}$), which is called as the Maximum Likelihood Decoding.

Lemma 1 *Over BSC as in Figure 1, MLD reduces to minimum distance decoding(MDD).*

Proof: Let $d_H(\vec{y}, \vec{c_i}) = d$.

P($\vec{y}|\vec{c_i}$) $= \varepsilon^d(1-\varepsilon)^{n-d} = (1-\varepsilon)^n(\frac{\varepsilon}{1-\varepsilon})^d$

If $\varepsilon < 0.5, \frac{\varepsilon}{1-\varepsilon} < 1$

$\implies$ P($\vec{y}|\vec{c_i}$) is maximized by minimizing $d_H(\vec{y}, \vec{c_i})$

In case of AWGN channel,

$$\text{P}(\vec{y}|\vec{c_i}) = \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp(\frac{-1}{2\sigma^2}\ (y_j - c_{ij})^2),$$

where c_{ij} is the j^th component of $\vec{c_i}$. Thus MLD reduces to minimum Euclidean distance decoding given by

$$d_E^2(y, c_i) = \sum_{i=1}^{n} (y_i - c_{ij})^2 \tag{2}$$

2 Syndrome Decoding(Slepian Wolf Decoding)

Consider BSC and let the code used be linear code. The rule followed in MLD is to find $\vec{c_i}$ such that $d_H(\vec{y}, \vec{c_i})$ is minimum.

$$d_H(\vec{y}, \vec{c_i}) = \{W_H(\vec{y} + \vec{c_i}) \mid \vec{c} \in \mathcal{C}\}$$

So, equivalently, we are looking for $\vec{c_i}$ such that,

$$W_H(\vec{y} + \vec{c_i}) = \min \{W_H(\vec{y} + \vec{c_j}) \mid \vec{c_j} \in \mathcal{C}\} = min_z\{W_H(\vec{z}) \ \mid \vec{z} \in \vec{y} + \mathcal{C}\}$$

Here, $\vec{y} + \mathcal{C}$ is a coset of code $\mathcal{C}$. Now having found such a $\vec{z}$, for some i, we have

$$\hat{c} = \vec{c_i} = \vec{y} + \vec{z}$$

Hence, the decoding algorithm is as follows
(a) Given $\vec{y}$, form $\vec{y} + \mathcal{C}$
(b) Look for least Hamming weight vector $\vec{z}$ in $\vec{y} + \mathcal{C}$
(c) $\hat{c} = \vec{y} + \vec{z}$

Lemma Let $\mathcal{C}$ be a [n,k] linear code and H be its p-c matrix. Then, there exists a one-to-one correspondence between the sets $\{\ \vec{y} + \mathcal{C}\}$ and $\mathbb{F}_2^{n-k}$, given by $H\vec{y} \in \mathbb{F}_2^{n-k}$.

Proof: Clearly collection of all $\vec{y_i} + \mathcal{C}$ is of size 2^{n-k}.

If $\vec{y_1} + \mathcal{C} = \vec{y_1}' + \mathcal{C}$
$\implies \vec{y_1} + \vec{c_1} = \vec{y_1}' + \vec{c_2}$
$\implies$ H$\vec{y_1}$ + H$\vec{c_1}$ = H$\vec{y_1}'$ + H$\vec{c_2}$
$\implies$ H$\vec{y_1}$ = H$\vec{y_1'}$

It remains to show that the mapping is H.

$$H\vec{y_1} = H\vec{y_2} \implies H(\vec{y_1} + \vec{y_2}) = 0$$

$\implies \vec{y_1} + \vec{y_2} \in \mathcal{C}$
$\implies \vec{y_1} = \vec{y_2} + \vec{c}\ ,\ \vec{c} \in \mathcal{C}$
$\implies \vec{y_1} + \mathcal{C} = \vec{y_2} + \mathcal{C}$

For example consider $\mathcal{C}$[n,k,d] = [4,2,2]

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

i.e. a self dual code

let $\mathcal{C} = \{\vec{c_1},\vec{c_2},\vec{c_3},\vec{c_4}\}$

$$c_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$c_2 = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$c_3 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

$$c_4 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Table 1: Standard Array Decoding

$\vec{c_1}^T$	$\vec{c_2}^T$	$\vec{c_3}^T$	$\vec{c_4}^T$	$\vec{s}^T$	
0 0 0 0	1 0 1 0	0 1 0 1	1 1 1 1	0 0	$\mathcal{C}$
0 0 0 1	1 0 1 1	0 1 0 0	1 1 1 0	0 1	$\mathcal{C}$ + 0 0 0 1
0 0 1 0	1 0 0 0	0 1 1 1	1 1 0 1	1 0	$\mathcal{C}$ + 0 0 1 0
0 0 1 1	1 0 0 1	0 1 1 0	1 1 0 0	1 1	$\mathcal{C}$ + 0 0 1 1

The first column in the above table is known as the Coset Leader. Also, We have

$$2t_c + 1 \leqslant d_{min} = 2 \implies t_c = 0$$

$$\text{Syndrome } \vec{s} = H\vec{y}$$

<u>Decoding Technique</u>

Suppose

$$y = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

$$Hy = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$e = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

$$c = y + e = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

Clearly, only those error patterns corresponding to coset leaders (first column of Table 1) can be correctly decoded.

2.1 Performance of Standard Array Decoder

Let $\vec{e}$ be the actual error pattern.

$$\therefore \vec{y} = \vec{c} + \vec{e}$$
$$\vec{s} = \text{H}\vec{y} = \text{H}(\vec{c} + \vec{e}) = \text{H}\vec{e}$$

$\therefore$ Syndrome associated with $\vec{y}$ is same as that associated with $\vec{e}$.
Let $\hat{e}$ be the coset leader associated with syndrome $\vec{s} = \text{H}\vec{e}$.

$$\hat{c} = \vec{y} + \hat{e} = \vec{c} + \vec{e} + \hat{e}$$

$\implies$ Decoding is correct only if $\vec{e} = \hat{e}$ i.e. iff the error pattern is coset Leader.

3 Reed Muller Codes

Reed Muller Codes are based on Boolean Functions. A Boolean function f in binary multivariables is a mapping

$$f : \begin{bmatrix} X_1 \\ X_2 \\ . \\ . \\ . \\ X_m \end{bmatrix} \to \mathbb{F}_2$$

For example the truth table as follows is an example of Boolean Function.

Table 2: Truth Table for X_1, X_2 and X_3

X_1	$X_2\ X_3 = 0\ 0$	$X_2\ X_3 = 0\ 1$	$X_2\ X_3 = 1\ 1$	$X_2\ X_3 = 1\ 0$
0	0	1	1	0
1	1	1	1	1

Clearly, from Table 2, 2^{2^n} boolean functions are possible with n variables (n = 3 in Table 2). Thus there exists a one-to-one mapping from set of all boolean functions to the set of all truth tables. i.e. given a boolean function, the corresponding truth table can be deduced.

Every boolean function has a unique representation as a multivariate polynomial in n-binary variables over $\mathbb{F}_2$ by Lagrange Interpolation as

$$f(X_1, X_2, ...X_m) = \sum_{a1}\sum_{a2} .... \sum_{am} f(a_1, a_2, ...a_m) \prod_{i=1}^{m}(X_i + a_i + 1)$$

This is called as the Reed Muller Canocial expansion of Boolean function. For example consider table 3 below.

Table 3: Truth Table for X_1, X_2 and X_3

X_1	$X_2\ X_3 = 0\ 0$	$X_2\ X_3 = 0\ 1$	$X_2\ X_3 = 1\ 1$	$X_2\ X_3 = 1\ 0$
0	0	1	0	0
1	0	0	1	0

$$g(X_1X_2X_3) = (X_1{+}1)(X_2{+}1)(X_3){+}X_1X_2X_3 = X_3{+}X_3X_1{+}X_3X_2{+}X_3X_1X_2{+}X_1X_2X_3$$

$$\implies g(X_1X_2X_3) = X_3 + X_1X_3 + X_2X_3$$

Clearly, the degree of the monomial $X_{i1}X_{i2}X_{i3}....\ X_{ir}$ is r. The degree of a Boolean function is the largest degree of a monomial in its Reed Muller Canonical Expansion.

For (X_1, X_2,.... X_m) $\in \{0,1\}^m$, we have $\sum_X f(X_1, X_2...X_m) = \sum_X(\sum_e a_eX^e) = \sum_e a_e(\sum_X X^e) = \{1$, iff $a_{1111} = 1$ and 0, elsewhere$\}$.

For example, $\sum_{X_1X_2X_3}(X_3{+}X_1X_2{+}X_2X_3) = \sum_{X_1X_2X_3} X_3{+}\sum_{X_1X_2X_3} X_1X_2{+} \sum_{X_1X_2X_3} X_2X_3 = 4 + 2 + 2 = 0$.