Lecture #3A: Syndrome, error correction and error detection
Outline of the lecture

- Syndrome and error detection

- Syndrome and error correction
Syndrome and error detection

Let $v = (v_0, v_1, \cdots, v_{n-1})$ be a codeword from a binary $(n,k)$ linear block code with generator matrix $G$ and parity check matrix $H$.

Assume $v$ is transmitted over a BSC, then binary received sequence, 

$$r = (r_0, r_1, \cdots, r_{n-1}) = v + e \pmod{2}$$

$$= (v_0, v_1, \cdots, v_{n-1}) + (e_0, e_1, \cdots, e_{n-1})$$

$$= (v_0 + e_0, v_1 + e_1, \cdots, v_{n-1} + e_{n-1})$$

where the binary vector $e = (e_0, e_1, \cdots, e_{n-1})$ is the error pattern.
Syndrome and error detection

- Let $v = (v_0, v_1, \cdots, v_{n-1})$ be a codeword from a binary $(n,k)$ linear block code with generator matrix $G$ and parity check matrix $H$.
- Assume $v$ is transmitted over a BSC, then binary received sequence,

$$r = (r_0, r_1, \cdots, r_{n-1}) = v + e \pmod{2}$$

$$= (v_0, v_1, \cdots, v_{n-1}) + (e_0, e_1, \cdots, e_{n-1})$$

$$= (v_0 + e_0, v_1 + e_1, \cdots, v_{n-1} + e_{n-1}),$$

where the binary vector $e = (e_0, e_1, \cdots, e_{n-1})$ is the error pattern.
- The “1’s” in $e$ represent transmission errors, i.e.,

$$e_i = \begin{cases} 
1 & \text{if } r_i \neq v_i \\
0 & \text{if } r_i = v_i,
\end{cases}$$

and $e_i = 1$ indicates that the $i^{th}$ position in $r$ has an error.

- After receiving $r$, the decoder must determine if $r$ contains errors (error detection), and locate the errors in $r$ (error correction).
Syndrome and error detection

- After receiving $\mathbf{r}$, the decoder must determine if $\mathbf{r}$ contains errors (error detection), and locate the errors in $\mathbf{r}$ (error correction).
- *Error detection* is achieved by computing the $(n-k)$ tuple
  \[
  \mathbf{s} = (s_0, s_1, \cdots, s_{n-k-1}) = \mathbf{r}^T \quad (\text{syndrome})
  \]

$r$ is a codeword if and only if $\mathbf{s} = \mathbf{r}^T = \mathbf{0}$. 
After receiving \( r \), the decoder must determine if \( r \) contains errors (error detection), and locate the errors in \( r \) (error correction).

*Error detection* is achieved by computing the \((n-k)\) tuple

\[
\mathbf{s} = (s_0, s_1, \cdots, s_{n-k-1}) = r^T \mathbf{H} \quad \text{(syndrome)}
\]

\( r \) is a codeword if and only if \( \mathbf{s} = r^T \mathbf{H} = 0 \).

If \( \mathbf{s} \neq 0 \), \( r \) is not a codeword and transmission errors have been detected.

If \( \mathbf{s} = 0 \), \( r \) is a codeword and no errors are detected. If \( r \) is a codeword other than the actual transmitted codeword, then an *undetected error* occurs. This happens whenever the error pattern \( \mathbf{e} \) is a non-zero codeword.
The syndrome $s$ computed from the received vector $r$ actually depends only on the error pattern $e$, and not on the transmitted code word $v$.

$$s = r \cdot H^T = (v + e)H^T = v \cdot H^T + e \cdot H^T$$

Since $v \cdot H^T = 0$,

$$s = e \cdot H^T$$
Syndrome and error detection

Example 2.4: Consider a \((7, 4)\) linear code with parity-check matrix

\[
H = \begin{bmatrix}
1 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 & 1 \\
\end{bmatrix}
\]

Let \(r = (0 1 0 0 0 0 1)\). The syndrome of \(r\) is

\[
s = (s_0, s_1, s_2) = r \cdot H^T = (0 1 0 0 0 0 1) \cdot \begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
1 & 1 & 0 \\
0 & 1 & 1 \\
1 & 1 & 1 \\
1 & 0 & 1 \\
\end{bmatrix} = (1 1 1) \neq 0
\]

Syndrome and error correction

- The syndrome \(s\) computed from the received vector \(r\) actually depends only on the error pattern \(e\), and not on the transmitted code word \(v\).

\[
s = r \cdot H^T = (v + e)H^T = e \cdot H^T \quad \text{(since } vH^T = 0)\]
The syndrome $s$ computed from the received vector $r$ actually depends only on the error pattern $e$, and not on the transmitted code word $v$.

$$s = r \cdot H^T = (v + e)H^T = e \cdot H^T \quad \text{(since } vH^T = 0)$$

For error pattern $e = \{e_0, e_1, \cdots, e_{n-1}\}$, and $H$ given by

$$H = [I_{n-k} : P]$$

$$= \begin{bmatrix}
1 & 0 & 0 & \cdots & 0 & p_0,0 & p_1,0 & \cdots & p_{k-1},0 \\
0 & 1 & 0 & \cdots & 0 & p_0,1 & p_1,1 & \cdots & p_{k-1},1 \\
0 & 0 & 1 & \cdots & 0 & p_0,2 & p_1,2 & \cdots & p_{k-1},2 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & 1 & p_0,n-k-1 & p_1,n-k-1 & \cdots & p_{k-1,n-k-1}
\end{bmatrix}$$

the syndrome equations can be rewritten as

$$s_j = e_j + e_{n-k}p_{0j} + e_{n-k+1}p_{1j} + \cdots + e_{n-1}p_{k-1,j}, \quad 0 \leq j \leq n - k$$

This is a set of $n - k$ equations in $n$ unknowns, $e_0, e_1, \cdots, e_{n-1}$.
Syndrome and error correction

- This is a set of $n - k$ equations in $n$ unknowns, $e_0, e_1, \ldots, e_{n-1}$.
- The decoder must solve of these equations for the estimated error pattern, $\hat{e}$. Estimated codeword is

$$\hat{v} = r + \hat{e}$$

- There are $2^k$ possible solutions to the syndrome equations and only one solution represents the true error pattern.
Syndrome and error correction

This is a set of $n - k$ equations in $n$ unknowns, $e_0, e_1, \cdots, e_{n-1}$.

The decoder must solve of these equations for the estimated error pattern, $\hat{e}$. Estimated codeword is

$$\hat{v} = r + \hat{e}$$

There are $2^k$ possible solutions to the syndrome equations and only one solution represents the true error pattern.

To minimize the probability of a decoding error, the \textit{the most probable} error pattern that satisfies the above equations is chosen as the true error vector.

Recall for BSC, the maximum likelihood decoder choose $\hat{v}$ as the codeword $\hat{v}$ that minimizes Hamming weight of the error pattern $e$. 
Example 3.1: Let

\[
H = \begin{bmatrix}
  1 & 0 & 0 & 1 & 0 & 1 & 1 \\
  0 & 1 & 0 & 1 & 1 & 1 & 0 \\
  0 & 0 & 1 & 0 & 1 & 1 & 1
\end{bmatrix}
\]

Suppose \( v = (1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1) \) is transmitted and \( r = (1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 1) \) is received. Then the syndrome of \( r \) is

\[
s = (s_0, s_1, s_2) = r.H^T = (1 \ 1 \ 1)
\]

Let \( e = (e_0, e_1, e_2, e_3, e_4, e_5, e_6) \) be the error pattern.

Since

\[
s = e.H^T
\]

we have the following 3 equations:

\[
\begin{align*}
1 & = e_0 + e_3 + e_5 + e_6 \\
1 & = e_1 + e_3 + e_4 + e_5 \\
1 & = e_2 + e_4 + e_5 + e_6
\end{align*}
\]
The solutions are:

\[
\begin{align*}
(0000010) & \quad (1010011) \\
(1101010) & \quad (0111011) \\
(0110110) & \quad (1100111) \\
(1011110) & \quad (0001111) \\
(1110000) & \quad (0100001) \\
(0011000) & \quad (1001001) \\
(1000100) & \quad (0010101) \\
(0101100) & \quad (1111101)
\end{align*}
\]

Note that the true error pattern,

\[e = r + v = (1001001) + (1001011) = (0000010)\]

is one of the 16 possible solutions. It is also the most probable solution.