Tut 6. Ex 1

Uncoded case:

bit error probability \( P_e = G \left( \frac{2E_b}{N_R} \right) \cdot Q \left( \frac{1}{2} \right) = \frac{3}{9} \cdot 10^{-6} \)

probability of making an error in a message (12 bits):

\[ 1 - (1 - P_e)^{12} = 9.65 \cdot 10^{-5} \]

Coded case:

To have the same \( E_b/N_0 \), we must double the rate.

Hence \( E_b/N_0 = 2 \sigma B = 5 \)

bit error probability for the coded bits \( P_e = G \left( \sqrt{10} \right) = 7.8 \cdot 10^{-6} \)

probability of making a block error:

\[ \sum_{n=4n}^{2(n+1)} \binom{n}{k} P_e^k (1 - P_e)^{n-k} \left[ n = 2n, k = 2 \right] = 9.65 \cdot 10^{-7} \]

(using only the first term gives the value \( 9.55 \cdot 10^{-7} \))

\[ \frac{9.65 \cdot 10^{-5}}{9.65 \cdot 10^{-7}} = 98.2 \]
Tut 6. Ex 2

This system transfers bits faster than we require, but the BER over the transmission is too poor. Now we introduce a simple coding scheme in which every bit (now regarded as coded bits) is repeated five times. Hence the information bit rate is reduced fivefold, but what is the new BER?

Three or more bit errors on the transmission will cause a bit error after the decoder. \( P_e = 10^{-2} \).

\[
P_e = \left( \frac{5}{2} \right) p_e^5 + \left( \frac{5}{4} \right) p_e^4 (1-p_e) + p_e^5 \approx 9.9 \times 10^{-9}
\]

(using only the first term gives \( P_e = 9.98 \times 10^{-9} \))

Daniel Houston
Theorem 6. Existence Theorem

\[ g(x) = 1 + x + x^2 + x^5 + x^6 + x^{10} \]

1) \[ m(x) = 1 + x^2 + x^4 \]

\[ x^m(x) = x^{10} \left( 1 + x^2 + x^4 \right) = x^{10} + x^{12} + x^{14} = q(x)g(x) + r(x) \]

\[ r(x) \]

\[ \text{codeword} = x^{n-m}m(x) + r(x) = 1 + x + x^2 + x^5 + x^6 + x^7 + x^9 + x^{10} + x^{15} + x^{19} \]

2) A codeword is divisible by \( g(x) \):

\[ \text{codeword} = x^{n-m}m(x) + r(x) = q(x)g(x) + r(x) + r(x) = q(x)g(x) + r(x) \]

\[ V(x) = 1 + x^4 + x^6 + x^8 + x^{14} \]

\[ x^9 + x^4 + 1 \]

\[ x^{10} + x^6 + x^7 + x^9 + 1 \]

\[ x^{10} + x^6 + x^7 + x^9 + 1 \]

\[ x^{10} + x^6 + x^7 + x^9 + 1 \]

\[ x^{10} + x^6 + x^7 + x^9 + 1 \]

\[ x^{10} + x^6 + x^7 + x^9 + 1 \]

\[ \Rightarrow V(x) \text{ is not a codeword} \]

Daniel Moneim
Tut 6, Ex 3

\[ G = \begin{bmatrix}
1 & 1 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 \\
1 & 1 & 0 & 0 & 0 & 0
\end{bmatrix} \quad k = 9 \\
\begin{array}{c}
\hline
p \\
1 \\
\end{array} \quad n = 7

1) \quad M \quad \quad \quad X = MG

\[
\begin{array}{cccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{array}
\]

2) A codeword is also the difference between two codewords

**Proof:** \( X_1 - X_2 = (M_1 - M_2)G = M_G G = X_3 \)

Hence, look at the Hamming weights of the codewords

\( d_{\text{min}} = 3 \) (0000000 is the difference between a codeword and itself, so that doesn't count)

\( \Rightarrow \) The code can correct 1 error

3) The code can detect 2 errors.
4) \[ H^T = \begin{bmatrix} I_{n-k} & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \]

5) \[ t = 1 \Rightarrow \text{Consider single error pattern} \]

<table>
<thead>
<tr>
<th>Error Pattern (E)</th>
<th>Syndrome (S = EH^T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000001</td>
<td>110</td>
</tr>
<tr>
<td>00000100</td>
<td>011</td>
</tr>
<tr>
<td>00001000</td>
<td>101</td>
</tr>
<tr>
<td>00100000</td>
<td>111</td>
</tr>
<tr>
<td>01000000</td>
<td>001</td>
</tr>
<tr>
<td>10000000</td>
<td>010</td>
</tr>
<tr>
<td>10000000</td>
<td>100</td>
</tr>
</tbody>
</table>

6) \[ r = [1, 1, 0, 1, 1, 0, 1] \]

\[ S = rH^T = [0, 1, 0] \Rightarrow S \neq 0 \text{ (not a codeword!)} \]

Syndrome table says \( E = [0, 0, 0, 0, 0, 0] \)

\( \Rightarrow \) probably \( X = [1, 0, 0, 1, 1, 0, 1] \) \( \Rightarrow M = [1, 1, 0, 1] \)

[Signature: Daniel Janson]
1. 
\[ P_0 = Q(\sqrt{\frac{E_b}{N_0}}) = 6 \times 10^{-3} \]
\[ P_w = 1 - (1 - P_0)^2 = 0.0409 \]

2. If no retransmissions are required then the throughput would be

\[ \frac{R}{T} = R_e = 4.667 \text{ kbps}, \text{ if on average we perform } \frac{R_e}{T} \text{ (re)transmissions,} \]

then the throughput is \( \frac{R_e}{T} \). But what is \( \frac{R_e}{T} \)?

\[ T_n = \text{number of (re)transmissions} \]
\[ P_E = \text{probability of a word error} \]
\[ P_c = \text{probability of a correct word} = 1 - P_E \]

\[ \frac{R_e}{T} = E[T_n] = 1 \cdot P_c + 2 \cdot P_E \cdot P_c + 3 \cdot P_E^2 \cdot P_c + 4 \cdot P_E^3 \cdot P_c + \ldots = \sum_{n=1}^{\infty} n \cdot P_E^{n-1} (1 - P_E) \]

Differentiating

\[ \sum_{n=0}^{\infty} a^n = \frac{1}{1-a} \Rightarrow \sum_{n=0}^{\infty} n \cdot a^{n-1} = \frac{1}{(1-a)^2} = \sum_{n=1}^{\infty} n \cdot a^{n-1} \]

\[ \Rightarrow \frac{R_e}{T} = (1 - P_E) \cdot \frac{1}{(1 - P_E)^2} = \frac{1}{1 - P_E}, \quad P_E = 1 - (1 - P_0)^2 = 0.0857 \Rightarrow \frac{R_e}{T} = 1.0937 \]

\( \Rightarrow \) Throughput is 9267 kbps.