1. (25pts) Consider the linear block code whose parity check matrix is given by:

\[ H = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix} \text{, } (n-k) \times 3 \]

(a) (6pts) Based on the given \( H \), what is the minimum distance \( d_{\text{min}} \) of this code? How many random errors can this code correct? \[ d_{\text{min}} = 3 \]

\[ t = \left\lfloor \frac{d_{\text{min}} - 1}{2} \right\rfloor = \left\lfloor \frac{3 - 1}{2} \right\rfloor = 1 \]

(b) (8pts) Write down all coset leaders and their corresponding syndrome WITHOUT having to find the standard array.

(c) (6pts) Decode the vector 00111. Is the answer unique? If not, find all possible answers.

(d) (5pts) Suppose this code is used for transmission over a binary symmetric channel with parameter \( p \). What is the probability of decoding error using the syndrome decoding method?

\[ P(\epsilon) = 1 - \sum \alpha_i p^i (1-p)^{k-i} \]

where \( \alpha_i = \# \text{coset leaders with weight } i \)

\[ p = 1 \]

\[ s = 2 \]

\[ P(\epsilon) = 1 - \left[ \alpha_0 p^5 (1-p)^5 + \alpha_1 p^4 (1-p)^4 + \alpha_2 p^3 (1-p)^3 \right] \]

\[ \alpha_0 = 1 \]

\[ \alpha_1 = 5p \]

\[ \alpha_2 = 2p^2 \]

\[ P(\epsilon) = 1 - \left[ p^5 (1-p)^5 + 5p (1-p)^4 + 2p^2 (1-p)^3 \right] \]
2. (18 pts) Consider the systematic, recursive convolutional encoder shown below:

(a) (12 pts) Complete the following table. Note that the last 4 bits in the u sequence are terminating bits.

<table>
<thead>
<tr>
<th>time t</th>
<th>u</th>
<th>w</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>v(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(b) (6 pts) Specify the generator polynomials \( G(D) = [G_1(D), G_2(D)] \) corresponding to the encoder above. Confirm the correctness of your results in (a) by directly computing \( v(1)(D) \) from \( u(D) \) and \( G(D) \).

\[
\begin{align*}
\eta(D) &= \begin{bmatrix}
1 & 1 + D^4 \\
-1 + D + D^2 + D^3 + D^9
\end{bmatrix}
\end{align*}
\]

\[
u(D) = 1 + D + D^4 + D^5 + D^6 + D^9
\]

\[
v^{(1)}(D) = \eta(D) G_2(D) = \frac{(1 + D + D^4 + D^5 + D^6 + D^9)(1 + D^4)}{1 + D + D^2 + D^3 + D^9}
\]

\[
\eta(D) = (1 + D + D^6 + D^8 + D^{10} + D^{13})(1 + D^2 + D^5 + D^9 + D^{10} + D^{13})
\]

\[
u(D) = (1 + D + D^6 + D^8 + D^{10} + D^{13})(1 + D^2 + D^5 + D^9 + D^{10} + D^{13})
\]

\[
\eta(D) = (1 + D + D^6 + D^8 + D^{10} + D^{13})(1 + D^2 + D^5 + D^9 + D^{10} + D^{13})
\]

\[
u(D) = (1 + D + D^6 + D^8 + D^{10} + D^{13})(1 + D^2 + D^5 + D^9 + D^{10} + D^{13})
\]

\[
u(D) = (1 + D + D^6 + D^8 + D^{10} + D^{13})(1 + D^2 + D^5 + D^9 + D^{10} + D^{13})
\]

\[
u(D) = (1 + D + D^6 + D^8 + D^{10} + D^{13})(1 + D^2 + D^5 + D^9 + D^{10} + D^{13})
\]
3. (20 pts) The probability of an event error for the maximum likelihood decoding of convolutional codes is given by

\[ P(E) \leq \sum_{d = d_{\text{free}}} A_d P_d. \]

(a) (4 pts) Explain the meaning of \( A_d \) and \( P_d \).

(b) (4 pts) For the BSC channel with parameter \( p \), compute \( P_3 \) exactly.

(c) (6 pts) Consider an AWGN channel with BPSK modulation, and binary-output quantization (hard decision). Explain how to convert this channel to a BSC channel and find the corresponding parameter \( p \). Your answer should be a function of \( E_s/N_0 \).

(d) (6 pts) For the BSC channel with parameter \( p \), we have \( P_3(E) \approx B_{\text{free}}^2 d_{\text{free}}^{p_{\text{free}}/2} \). Use this formula and the result in (c) to derive an approximated expression of \( P_3(E) \) for the AWGN channel with hard decision. Your answer should be a function of \( E_s/N_0 \) and the code rate \( R \). Compare this \( P_3(E) \) with that of the uncoded case to drive the coding gain. (You may use \( Q(x) \approx \frac{e^{-x^2/2}}{\sqrt{2}} \).)

a) \( A_d \) is the number of codewords of weight \( d \).

\( P_d \) is the probability that the first event error has weight \( d \).

This first event error corresponds to the decoder choosing an incorrect path through the trellis. This path gives the codeword of weight \( d \).

b) \( P_3 = \frac{3}{e^2} \left( \frac{3}{e} \right)^{2} (1-p)^{3-e} \approx (\frac{3}{2}) p^3 (1-p^3) + (\frac{3}{3}) p^3 . \)

\( e = 2 \)

c) BPSK: \( 1 \rightarrow SE_s \quad 0 \rightarrow -SE_s \quad (v_i = 1) \quad (v_i = -1) \quad (0). \)

tx signal \( s(t) = v_i \sqrt{2E_s/T} \cos(2\pi f_0 t) \quad 0 \leq t \leq T. \)

rx signal \( y_i(t) = s(t) + n(t) \quad n(t) \quad \text{is white Gaussian noise}. \)

Sampling: \( y_i = \int_{0}^{T} y_i(t) \sqrt{2E_s/T} \cos(2\pi f_0 t) \ dt \)

\( y_i = v_i E_s + n' \quad n' \quad N(0, E_s N_0) \)
The probability of error is given by:

\[ p = \frac{1}{2} Q\left(\sqrt{\frac{2E_s}{N_0}}\right) + \frac{1}{2} Q\left(\sqrt{\frac{4E_s + S_e}{N_0}}\right) \]

Where:

\[ \sigma^2 = \frac{N_0}{2} \quad V_{E_s} = O + S_e 
\]

\[ P = Q\left(\sqrt{\frac{2E_s}{N_0}}\right) \approx \frac{1}{2} e^{-\frac{1}{2}} \]

If \( H_1 \) is true \( \rightarrow 1 \) is decoded

If \( H_0 \) is true \( \rightarrow 0 \) is decoded.

\[ P_b(E) \sim B \cdot \exp\left(\frac{-\frac{E_s}{N_0}}{2}\right) \]

Uncoded case: \( P_b(E) = Q\left(\sqrt{\frac{2E_s}{N_0}}\right) \approx \frac{1}{2} e^{-\frac{1}{2}} \)

For large SNR, \( e^{-\frac{E_s}{2N_0}} \) dominates \( P_b(E) \).

\[ \therefore \text{coding gain} = 10 \log_{10} \left(\frac{R_d \cdot \text{SNR}}{2}\right) \text{dB} \]

This is because for coded case \( P_b(E) \sim e^{-\frac{R_d \cdot \text{SNR}}{2}} \) at high \( \frac{E_s}{N_0} \). For uncoded case \( P_b(E) \approx 2^{-\frac{E_s}{2N_0}} \).

\[ \therefore \text{coding gain} = \frac{R_d \cdot \text{SNR}}{2} = 10 \log_{10} \left(\frac{R_d \cdot \text{SNR}}{2}\right) \text{dB} \]
4. (15 pts) A (3, 1, 2) convolutional code is used for transmission over a BSC channel. The input sequence has length $Q$ including two terminating bits. The trellis of this code is given in the figure below. The received signal sequence $r$ is provided under the bottom row of the trellis.

Perform Viterbi decoding to find the most likely transmitted sequence. You should write the partial path metric above each state and clearly mark each eliminated branch with a "X".

\[ r = 101 \]

\[ 011 \]

\[ 101 \]

\[ 110 \]

\[ 001 \]

\[ 010 \]

\[ 101 \]

\[ \min_{\mathcal{H}} \quad d(n, v), \]

The sequence: \[111, 010, 001, 110, 011, 000, 000\]

If $p$, \[1111000000\]
\[
\begin{align*}
D_t^k(0, 0) &= \frac{-1}{2} (0.2) + \frac{2}{2} (-1(0.6) + (-1)(1.6) + (-1)(0.4)) = 0.1 + 0.4 = 0.5 \\
\alpha_t^k(0, 1) &= \frac{1}{2} (0.2) + (1)(0.9) + (1)(-1.0) + (1)(0.4) = 0.1 + 0.4 = 0.5
\end{align*}
\]

5. (22 pts) A (3, 1, 2) convolutional code is used for transmission over an AWGN channel with BPSK modulation such that 0 → -1 and 1 → +1. The signal-to-noise ratio \( \frac{E_b}{N_0} \) equals \( \frac{1}{2} \) (linear). A trellis section is shown in the figure below. The received signal vector for this trellis section is \( r = [0.8 -1.6 0.4] \).

Also, assume that the a priori LLR for bit \( u_t \) is \( L_a(u_t) = 0.2 \). The values of \( \alpha_t^k(S_t) \) and \( \beta^t(S_t) \) are shown in the figure.

\[\begin{align*}
\alpha_t^k &= \log(\alpha) \\
\beta_t^k &= \log(2\lambda)
\end{align*}\]

\(\beta_t^k = \log(\alpha) + \log(2)\)

\(\beta_t^k = \log(\alpha) + \log(2)\)

(a) (5 pts) Compute the branch metric \( \gamma^k(S_t, S_{t+1}) \) for each branch in this trellis section and write the value at the top of each branch.

(b) (4 pts) Compute \( \alpha_{t+1}^k(S_t) \).

(c) (4 pts) Compute \( \beta_t^k(S_t) \).

(d) (6 pts) Compute \( L(u_t) \), the output LLR for \( u_t \)

\[\begin{align*}
\alpha_t^k(S_t) &= \max^k \left[ 2.5 + 2.1, 1.4 + 0.3 \right] = \max^k \left[ 4.6, 1.1 \right] \\
\beta_t^k(S_t) &= \max^k \left[ -2.3, 1.3 \right] \\
L(u_t) &= \max^k \left[ 1.4 + 0.3 + (-2) \right] = \max^k \left[ 2.7 \right] - \max^k \left[ -0.1, 2.6, -1.9, 4.0 \right]\]
\]

(e) (3 pts) If all the \( \alpha \) values (not \( \alpha^k \) values) are scaled by 2, how will the output LLR change? Explain why.

(c) THE OUTPUT WILL NOT BE AFFECTED, IF YOU SCALE \( \alpha \) BY TWO, i.e.

\(2 \cdot \alpha \), THEN YOU END UP JUST ADDING 1 \( \ln(2) \) TO YOUR ORIGINAL \( \alpha^k \).

\[\hat{\alpha}^k = \ln(2\alpha) = \ln(2) + \ln(\alpha) = \ln(2) + \alpha^k\]