SCHOOL OF ELECTRONICS ENGINEERING (SENSE)

BECE313L: Information Theory and Coding

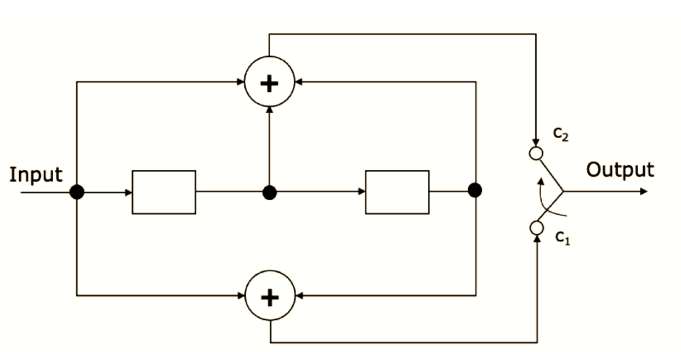
PROBLEM SHEET-4
Channel Coding : Part-2

Instructions:

1. Total Marks: 15
2. Weightage of marks in grades : 4%
3. Last Date for Submission: 21.11.2024
4. All answers must be handwritten
5. Late submission are not allowed
6. Submission must be through teams

Address each problem with thorough analysis and detailed solutions.

Q.No	Question	Marks
1	Consider a received sequence $r = 11111111001011$ from a $\frac{1}{2}$ convolutional encoder. Decode the given received vector using the Viterbi algorithm. The encoder circuit is provided below:	1



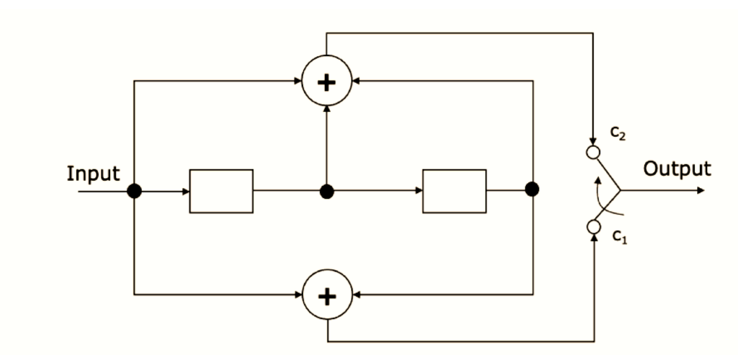
Provide detailed steps of the decoding process, including:

1. Construction of the trellis diagram.
2. Path metrics calculation for each step in the trellis.
3. Identification of the optimal path through the trellis and corresponding decoded bits.

2 For a $(2,1,2)$ convolutional encoder, find the codeword for the input sequence $u = [1,1,0,1,1,0]$ using the state diagram approach. 1

1. Construct the state diagram for the encoder.
2. Encode each bit of the input sequence u by following the transitions in the state diagram.
3. Write down the resulting codeword for the entire input sequence.

3 For the given $\frac{1}{2}$ convolutional encoder (shown below), encode the message 1010 using the state diagram approach. If the received codeword is 1101010010, decode the original message using the Viterbi algorithm. 1



Perform the following steps:

1. Encode the message 1010 and provide the resulting codeword.
2. Decode the received codeword 1101010010 using the Viterbi algorithm, including trellis construction and path metrics calculation.
3. Identify the decoded message and compare it to the original message.

4 For the given $(2, 1, 3)$ convolutional encoder, perform the following tasks: 2

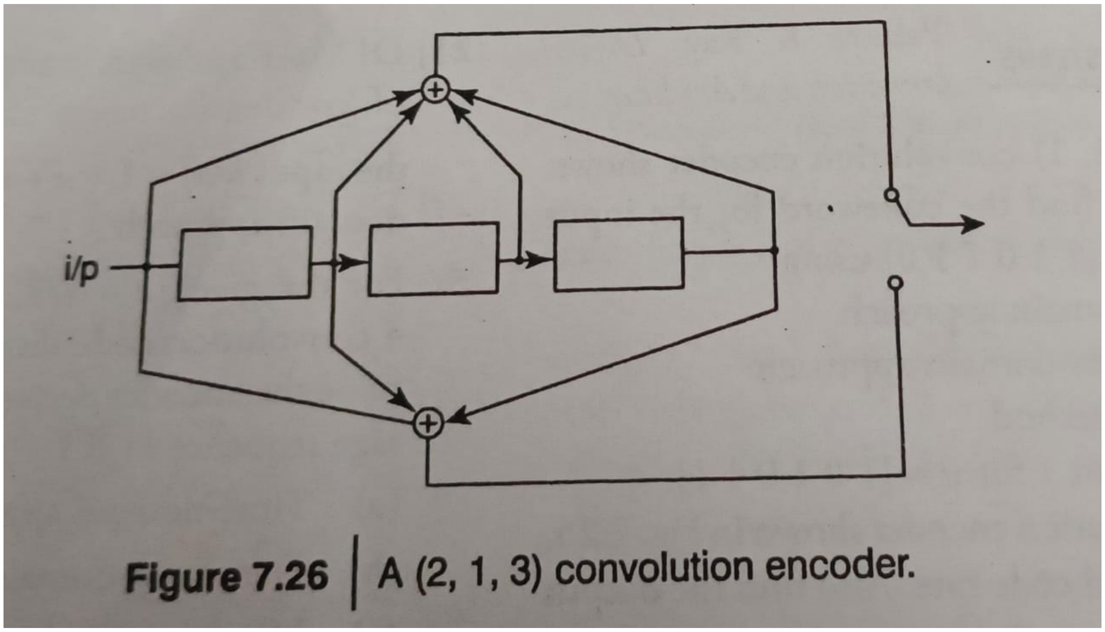


Figure 7.26 | A (2, 1, 3) convolution encoder.

1. Encode the input sequence $u = [1, 0, 1, 1, 0, 1, 0]$ using this encoder.
2. Draw the state diagram for this encoder and label all transitions.
3. If the received sequence is $r = 101110010111$, use the Viterbi algorithm to decode this sequence back to the original input message.
4. Analyze the error detection and correction capabilities of this encoder based on the state diagram and trellis paths.

5 Given the convolutional encoder, which has three memory elements (D_1, D_2, and D_3), with two outputs c_1 and c_2, answer the following: 2

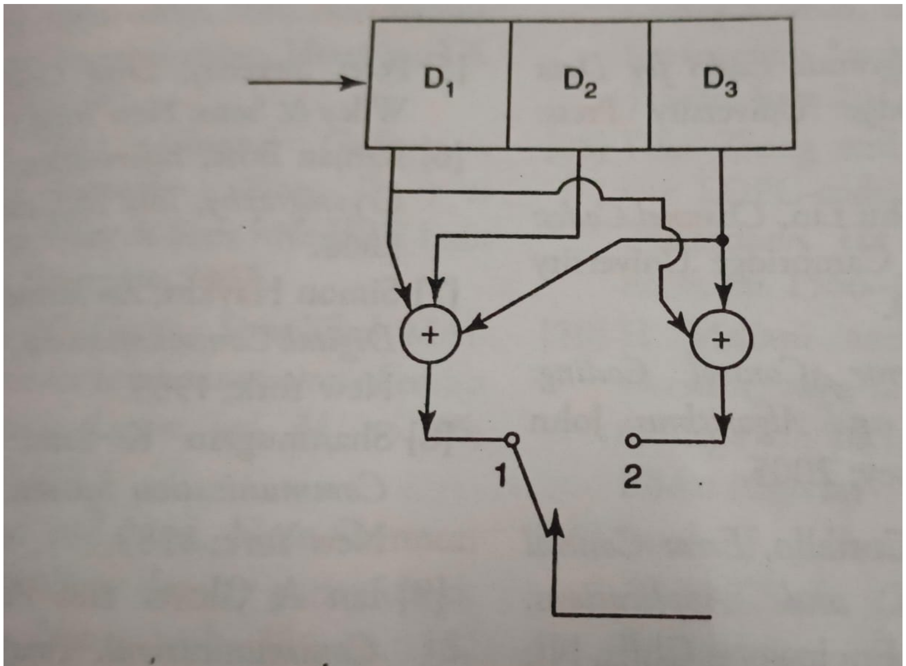


Figure 7.25 | Convolution encoder.

1. Encode the message $m = [1, 1, 0, 1, 0]$ and find the resulting codeword sequence.

2. Construct the trellis diagram for this encoder and label each branch with the corresponding output sequence.

3. If the received sequence is $r = 110110111001$, apply the Viterbi algorithm to decode it and retrieve the most likely original input.

4. Explain how the encoder's structure affects its minimum Hamming distance and error-correcting capability.

6 For the $(3, 2, 1)$ convolutional encoder depicted, perform the following tasks: 2

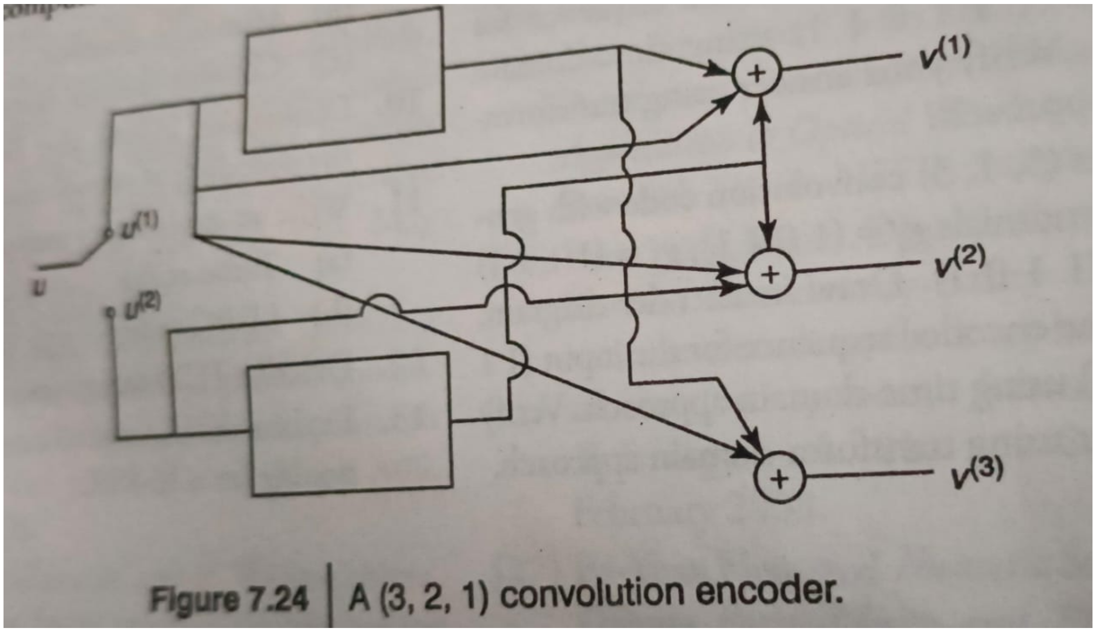


Figure 7.24 | A (3, 2, 1) convolution encoder.

1. Generate the codeword for the input sequence $u = [1, 0, 1, 1, 1, 0, 0]$ using the state transition approach.
2. Draw the state diagram for this encoder and identify all possible states and transitions.
3. If the received codeword is $r = 110101011100$, use the Viterbi algorithm to determine the original input sequence.
4. Discuss how changing the generator polynomial would affect the encoding process and the error-correcting capabilities of this encoder.

7 Consider a $(2, 1, 3)$ convolutional encoder represented in Figure 7.26 below. 1
Suppose the encoder is used to transmit the message $m = [1, 0, 0, 1, 1]$. Answer the following:

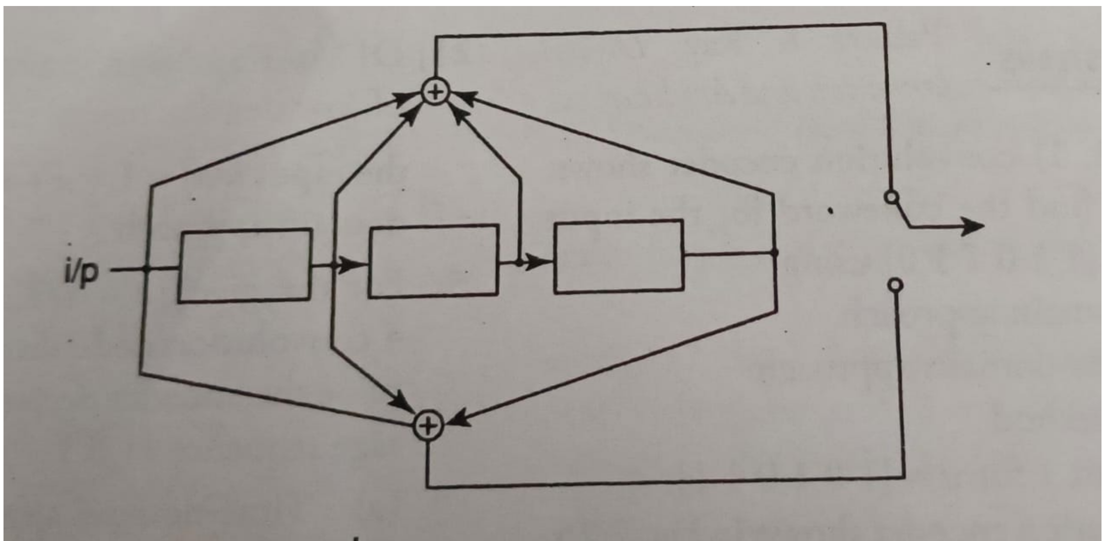


Figure 7.26 | A (2, 1, 3) convolution encoder.

1. Encode the message and provide the resulting output sequence.
2. Draw the corresponding trellis diagram for this encoder.
3. If the received sequence is $r = 101101011110$, apply the Viterbi algorithm to decode this sequence and find the estimated transmitted message.
4. Calculate the minimum Hamming distance for this convolutional code and determine its theoretical error detection and correction capabilities.

8 For the convolutional encoder shown in Figure below, which is a $\frac{1}{2}$ rate encoder: 1

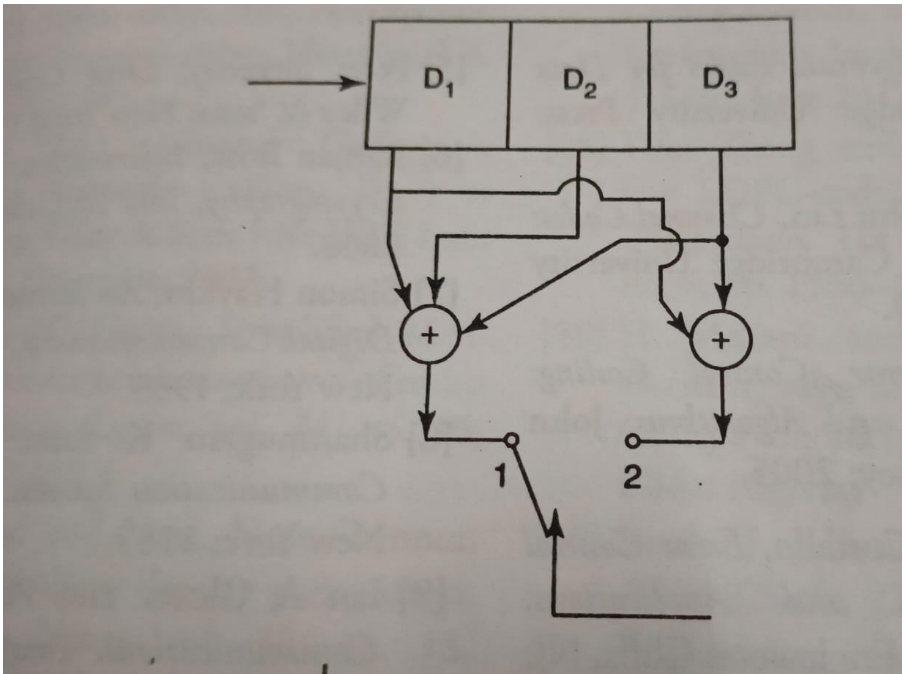


Figure 7.25 | Convolution encoder.

1. Encode the input sequence $u = [1, 1, 0, 0, 1]$ and give the resulting encoded output sequence.
2. Construct a state diagram for the encoder, showing all possible states and transitions.
3. If the codeword received is $r = 111001100110$, perform decoding using the Viterbi algorithm and identify any errors.
4. Describe how the encoder's memory elements contribute to its error-correction capability and the overall redundancy of the encoded output.

9 The parity check matrix for an $(8, 4)$ LDPC code is given below. Construct the Tanner graph for this code. 1

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

10 Consider the following parity check matrix: 1

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

Determine the minimum distance of the code and identify the category of the LDPC code.

11 Given a parity check matrix H for an LDPC code with a column weight of 3 and a row weight of 6: 1

1. Determine if this code qualifies as a regular LDPC code.
2. Construct a sample Tanner graph for a possible H matrix with these weights.
3. Discuss the error-correction capabilities of this regular LDPC code based on the column and row weights.

12 An LDPC code is defined by a parity check matrix H where the row weights vary while the column weights are fixed at 4. Identify this LDPC code as regular or irregular and justify your answer. Then, calculate the approximate decoding complexity of this code for a message length of 500 bits when using a belief propagation algorithm. 1