Quiz EL406
Channel Coding: Convolutional Codes
Lucas Monteiro Nogueira

PROBLEMS

Problem 1 (Proakis and Salehi, 2008, w/ permission)

A convolutional code is described by

$$g_1 = [101]; \quad g_2 = [111]; \quad g_3 = [111]$$

Problem 1.1: Draw the encoder corresponding to this code.
Problem 1.2: Draw the state-transition diagram for this code.
Problem 1.3: Draw the trellis diagram for this code.
Problem 1.4: Find the transfer function and the free distance of this code.
Problem 1.5: Verify whether or not this code is catastrophic.
Problem 1.6: The convolutional code introduced in this problem is used for transmission over an AWGN channel with hard decision decoding. The output of the demodulator detector is (101001011110111 ...). Using the Viterbi algorithm, find the transmitted sequence, assuming that the convolutional code is terminated at the zero state.

Problem 2 (Proakis and Salehi, 2008, w/ permission)

Consider now a code with

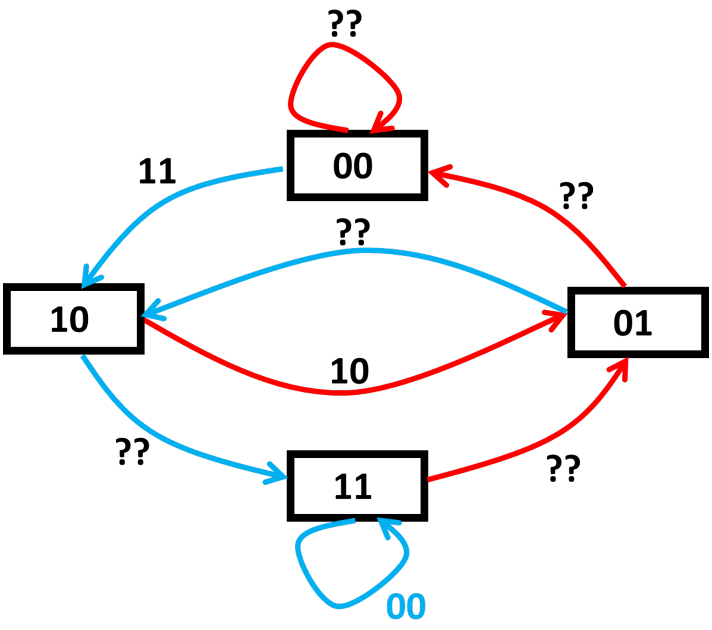
$$g_1 = [110]; \quad g_2 = [101]; \quad g_3 = [111]$$

Problem 2.1: Draw the encoder corresponding to this code.
Problem 2.2: Draw the state-transition diagram for this code.
Problem 2.3: Draw the trellis diagram for this code.
Problem 2.4: Find the transfer function and the free distance of this code.
Problem 2.5: Verify whether or not this code is catastrophic.

Problem 3 (Sklar, 2001, w/ permission)

Given a K = 3, rate 1/2, binary convolutional code with the partially completed state diagram shown below, find the complete state diagram and sketch the encoder circuit.

Note: *A red arrow denotes a "0" input; a blue arrow denotes a "1" input.*



Problem 4 (Proakis and Salehi, 2008, w/ permission)

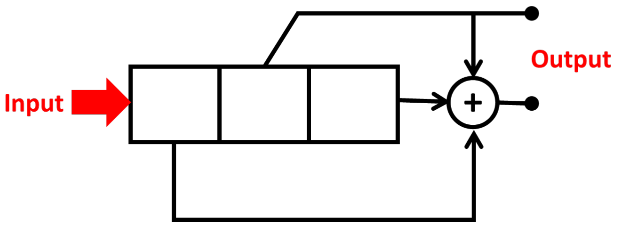
The block diagram for a convolutional code is given below.

Problem 4.1: Draw the state transition diagram for this code.

Problem 4.2: What is the transfer function for this code?

Problem 4.3: What is the free distance of this code?

Problem 4.4: Assuming that this code is used for binary data transmission over a binary symmetric channel with crossover probability of 10^{-3}, find a bound on the resulting bit error probability.

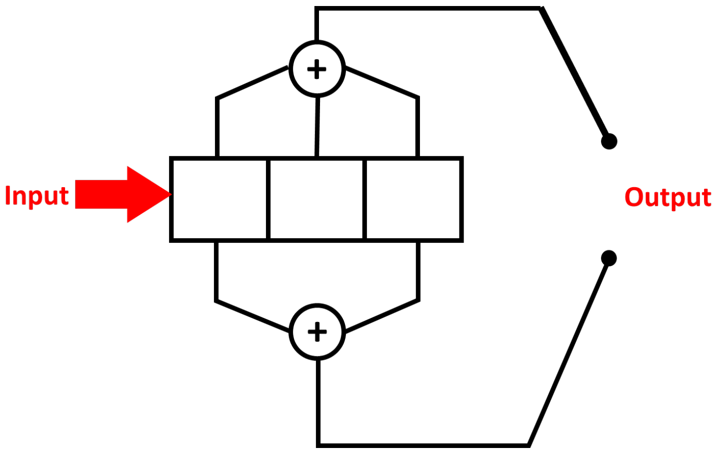


Problem 5 (Proakis and Salehi, 2008, w/ permission)

A rate 1/2, $K = 3$, binary convolutional encoder is shown below.

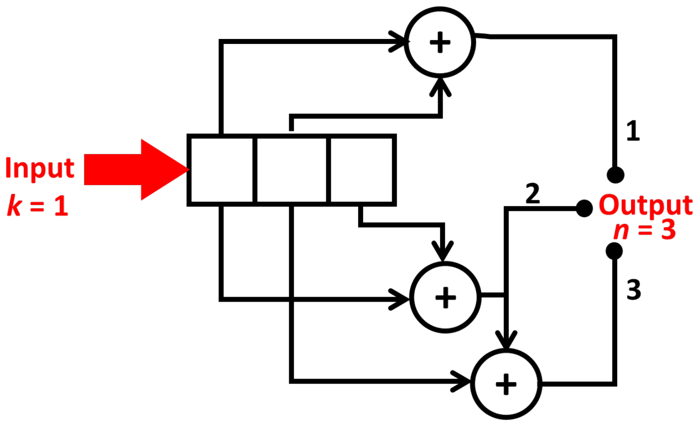
Problem 5.1: Draw the state diagram, the trellis diagram, and the tree diagram.

Problem 5.2: Determine the transfer function *T(D,N,J)*, and from this, specify the minimum free distance.



Problem 6 (Proakis and Salehi, 2008, w/ permission)

The block diagram of a (3,1) convolutional code is illustrated below.



Problem 6.1: Draw the state diagram of the code.

Problem 6.2: Find the transfer function of the code.

Problem 6.3: Find the minimum free distance d_f of the code.

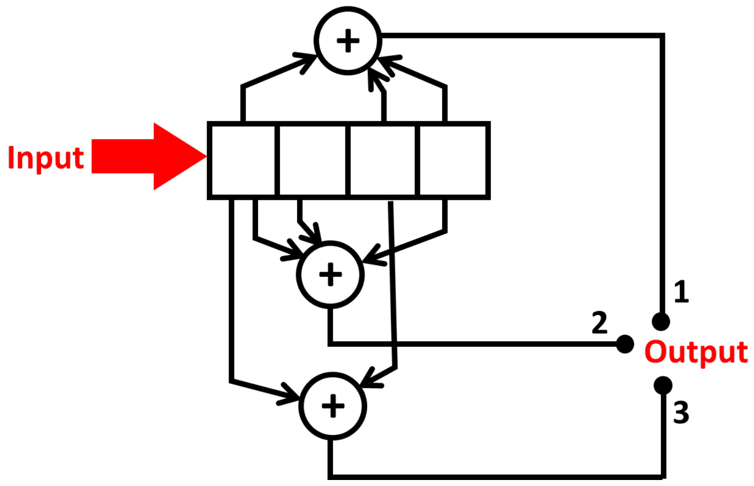
Problem 6.4: Assume that four information bits (x_1, x_2, x_3, x_4) followed by two zero bits have been encoded and sent via a binary-symmetric channel with crossover probability equal to 0.1. The received sequence is (111, 111, 111, 111, 111, 111). Use the Viterbi decoding algorithm to find the most likely data sequence, assuming that the convolutional code is terminated at the zero state.

Problem 7 (Sklar, 2001, w/ permission)

Table 1 in the additional information section gives the optimum short constraint length convolutional codes with rate 1/2 or 1/3. Using the table, devise a $K = 4$, rate 1/2 binary convolutional encoder. Draw the circuit and provide a trellis diagram.

Problem 8 (Sklar, 2001, w/ permission)

Consider the rate 2/3 convolutional encoder shown in the figure below. In this encoder, $k = 2$ bits at a time are shifted into the encoder and $n = 3$ bits are generated at the encoder output. There are $kK = 4$ stages in the register, and the constraint length is $K = 2$ in units of 2-bit bytes. The state of the encoder is defined as the contents of the rightmost $K - 1$ k-tuple stages. Draw the state diagram and the trellis diagram.



ADDITIONAL INFORMATION

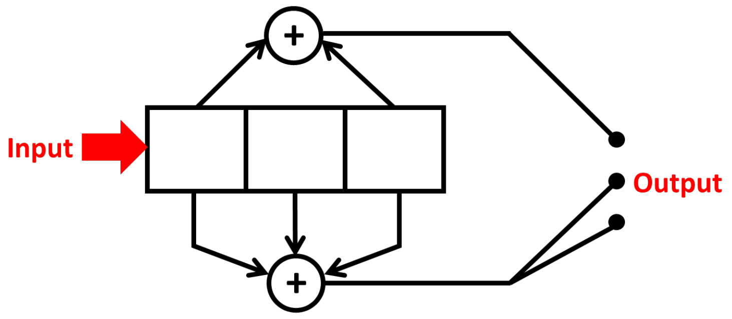
Table 1. *Optimum short constraint length convolutional codes (rate 1/2 and 1/3).*

Rate	Constraint length	Free distance	Code vector
1/2	3	5	111 101
1/2	4	6	1111 1011
1/2	5	7	10111 11001
1/2	6	8	101111 110101
1/2	7	10	1001111 1101101
1/2	8	10	10011111 11100101
1/2	9	12	110101111 100011101
1/3	3	8	111 111 101
1/3	4	10	1111 1011 1101
1/3	5	12	11111 11011 10101
1/3	6	13	10111 110101 111001
1/3	7	15	1001111 1010111 1101101
1/3	8	16	11101111 10011011 10101001

SOLUTIONS

P.1 Solution

Problem 1.1: The encoder for the code at hand is illustrated below.



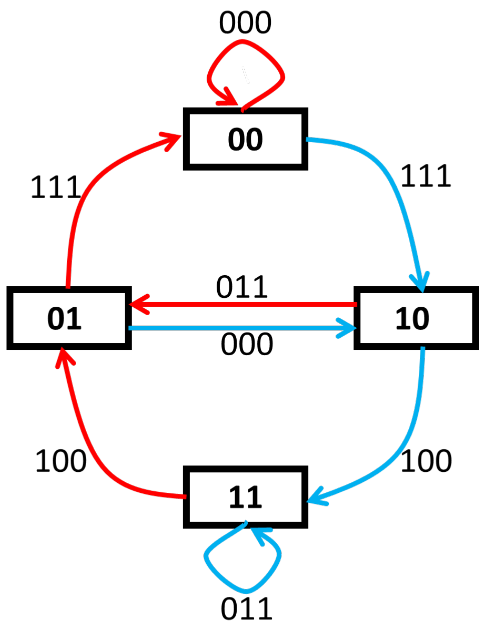
Problem 1.2: To draw the state-transition diagram, we first label states with four letters *A*, *B*, *C*, and *D*, as shown.

State	Label
00	*A*
01	*B*
10	*C*
11	*D*

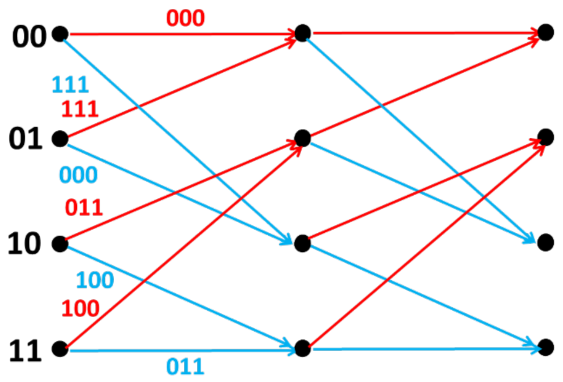
The code operations can be described by the equations $x_1 = m \oplus m_2$ and $x_2 = x_3 = m \oplus m_1 \oplus m_2$. We proceed to draw up the following table.

Input, m	m_1	m_2	Outputs			Current state	Next state
			x_1	x_2	x_3		
0	0	0	0	0	0	*A*	*A* 000 → 000
1	0	0	1	1	1	*A*	*C* 100 → 010
0	0	1	1	1	1	*B*	*A* 001 → 000
1	0	1	0	0	0	*B*	*C* 101 → 010
0	1	0	0	1	1	*C*	*B* 010 → 001
1	1	0	1	0	0	*C*	*D* 110 → 011
0	1	1	1	0	0	*D*	*B* 011 → 001
1	1	1	0	1	1	*D*	*D* 111 → 011

We now have all the information needed to produce the state transition diagram, which is shown below. **In this state diagram and the ones in subsequent problems, a red arrow denotes a "0" input and a blue arrow denotes a "1" input.**



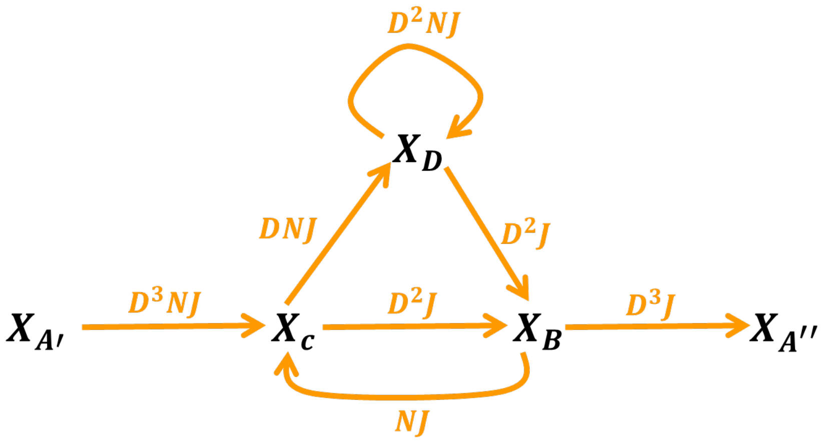
Problem 1.3: Drawing the trellis diagram is just as easy. Again, a red arrow denotes a "0" input and a blue arrow denotes a "1" input.



Problem 1.4: To find the transfer function, we evoke the general expression,

$$T(D,N,J) = \sum_{d=d_{\text{free}}}^{\infty} a_d D^d N^{f(d)} J^{g(d)}$$

Here, *d* is the number of ones in the output code word, *f(d)* is the number of ones in the input block *k*-bits, and *g(d)* is the No. of branches spanned by the path. Obtaining the transfer function is made much easier if we already have the state diagram, as in the present case. The transfer function diagram is sketched below.



To obtain the system from which the transfer function will be extracted, add the terms that arrive at each node of the flow graph. For example, there are two arrows arriving at X_C, one from $X_{A'}$ (associated with the term D^3NJ) and another from X_B (associated with the term NJ); accordingly, we may write

$$X_C = D^3NJX_{A'} + NJX_B$$

Proceeding similarly with X_B, X_D, and $X_{A''}$, we have

$$X_B = D^2JX_C + DJX_D$$
$$X_D = DNJX_C + D^2NJX_D$$
$$X_{A''} = D^3JX_B$$

Eliminating X_B, X_C and X_D results in the transfer function

$$T(D,N,J) = \frac{X_{A''}}{X_{A'}} = \frac{D^8NJ^3\left(1+NJ-D^2NJ\right)}{1-D^2NJ\left(1+NJ^2+J-D^2J^2\right)}$$

To find the free distance of the code we set $N = J = 1$ in the transfer function above and expand, giving

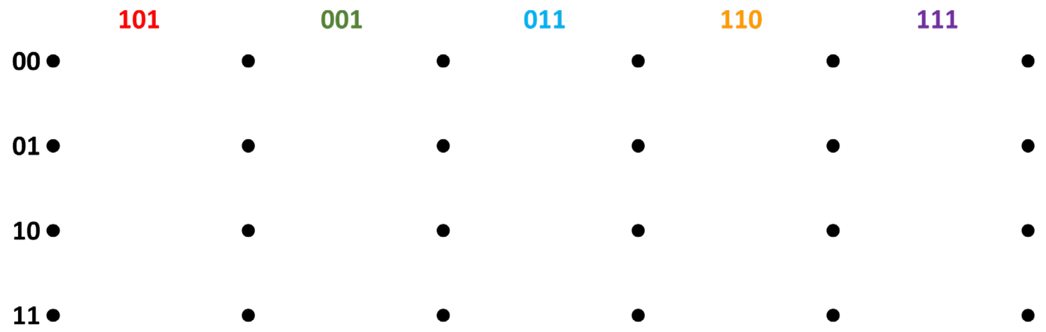
$$T_1(D) = T(D,N,J)\big|_{N=J=1} = \frac{D^8\left(1-2D^2\right)}{1-D^2\left(3-D^2\right)} = D^8 + 2D^{10} + \ldots$$

Clearly, $d_f = 8$.

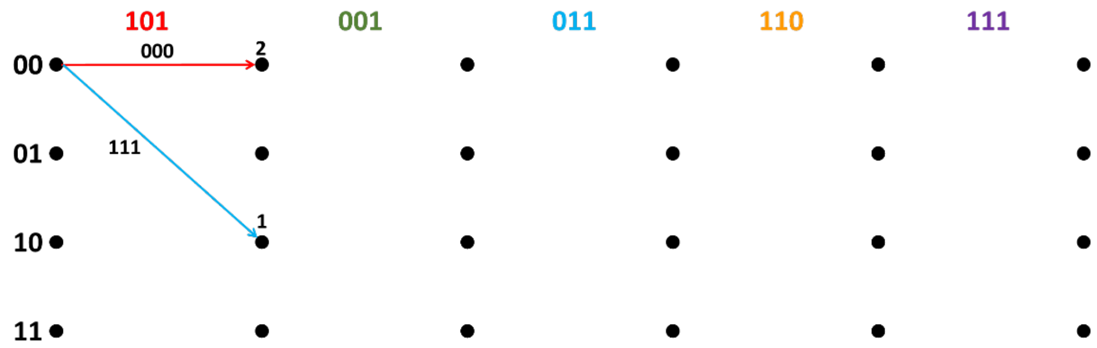
Problem 1.5: Since there is no self-loop corresponding to an input equal to 1 such that the output is the all zero sequence, the code is not catastrophic.

Problem 1.6: We first break down the message digits into groups of 3 and distribute them along the grid to be used for the extended trellis diagram.

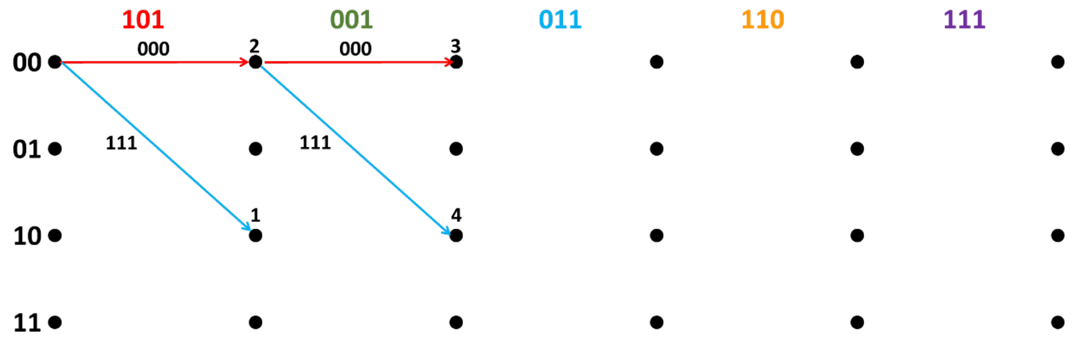
(101001011110111 ...)



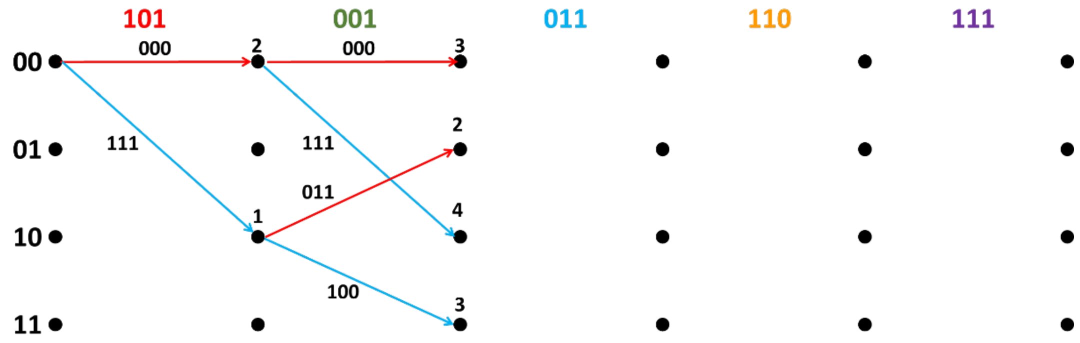
Referring to the state diagram presented in part 2, we see that the system can transition from 00 to 10 and produce 111 as output, or it can remain at 00 and yield 000 as output. In the former case, the Hamming distance from the 111 output relatively to 101 is 1, because the two codes differ in one bit. In the latter case, the Hamming distance from the 000 output relatively to 101 is 2, because the two codes differ in two bits. In the following illustration, note that we are using red and blue arrows to depict inputs of 0 and 1, respectively.



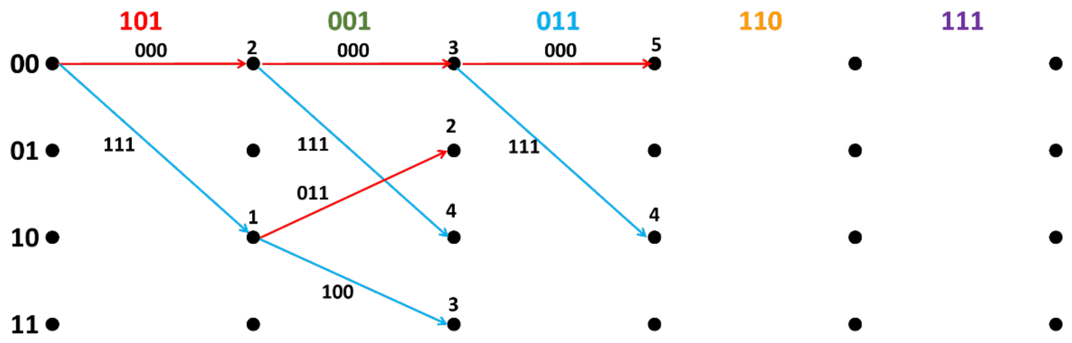
Consider now the second branch of the trellis. Starting at state 00 in the second column, the system can transition to 10 and yield 111 as the output, or it can remain at 00 and output 000. In the former case, the Hamming distance from the 111 output relatively to the 001 message triple is 2, which must be added to the 2 at the node we started at to give a 4. In the latter case, the output 000 differs from the 001 message triple by one bit, so we add a 1 to the 2 at the starting node to obtain a 3. The diagram is updated accordingly.



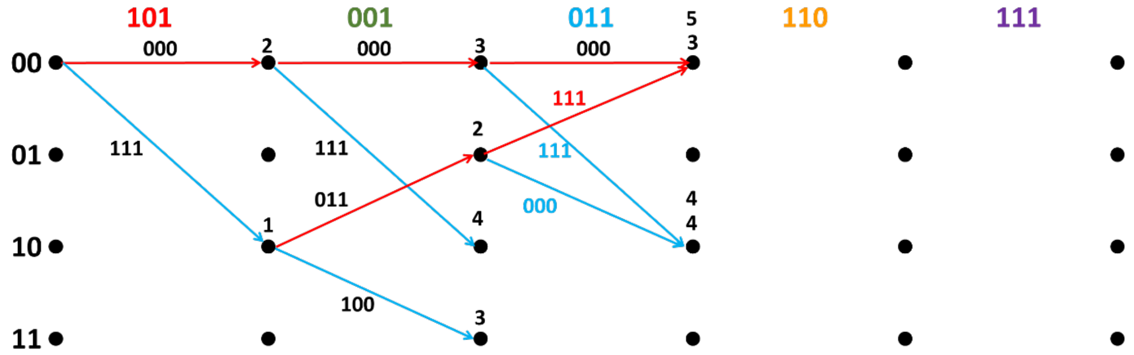
We must also update the paths stemming from the lower branch in the second column. Here, we start at state 10. As can be seen from the state diagram, we may transition to 01 while outputting a 011, which differs from the message triple 001 by one bit; this distance is added to the 1 we started with to yield a 2. Alternatively, we can transition from 01 to 11 while outputting a 100, which differs from the message triple 001 by two bits, and is then added to the 1 we started with to yield a 3. The diagram is updated accordingly.



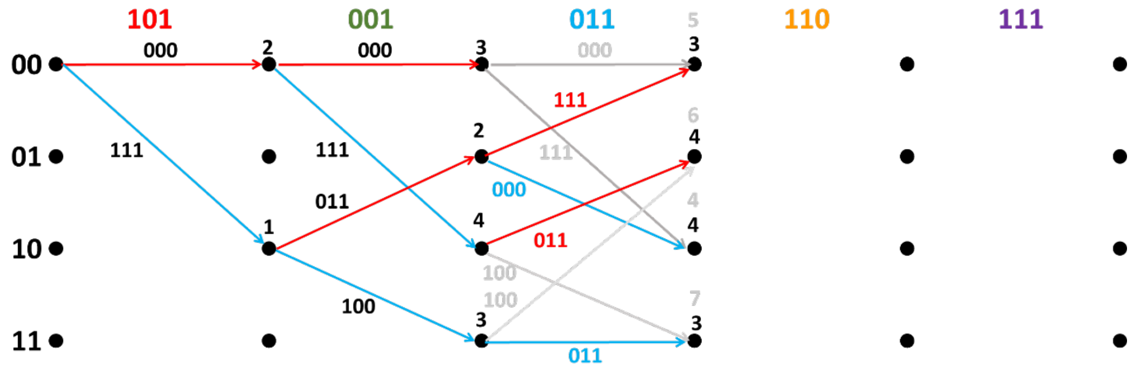
Let us inspect the third column. Starting from the 00 node again, the system can transition to 10 and yield 111 as the output, or it can remain at 00 and output 000. In the former case, the Hamming distance from the 111 output relatively to 011 is 1, which is added to the 3 we started with to yield an updated metric of 4. In the latter case, the Hamming distance from the 000 output relatively to 011 is 2, which is added to the 3 we started with to yield an updated metric of 5. The diagram is updated accordingly.



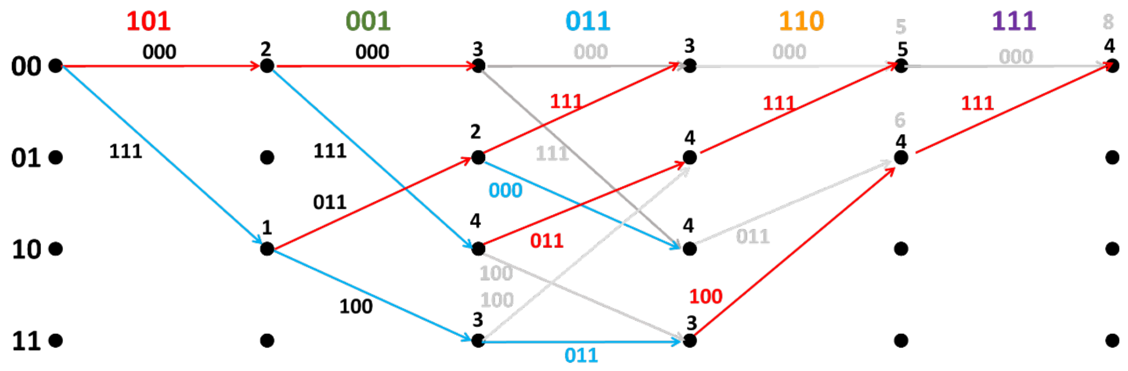
Now, suppose we take over from the 01 state. With reference to the state diagram, we can transition to 00 and yield an output of 111 or transition to 10 and yield an output of 000. In the former case, the 111 output differs from the 011 message triple in one bit, so a 1 is added to the 2 we started with to give a metric of 3. In the latter case, the 000 output differs from the 011 message triple in two bits, so a 2 is added to the 2 we started with to give a metric of 4. The diagram is updated accordingly.



At this point, we have more than one path arriving at the same node. In accordance with the Viterbi algorithm, in each node with more than one path pointing to it, we exclude the path with the larger metric. Notice that in state 10 there are two paths with the same metric, namely 4; the usual way to proceed in such cases is to purge the upper path, as shown.



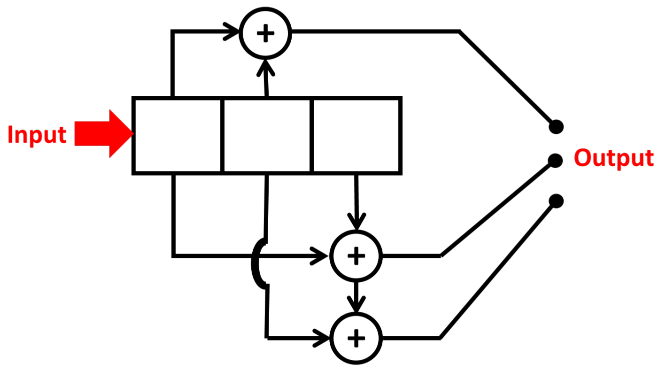
The remainder of the trellis diagram is sketched in continuation.



The surviving path follows the route 00 → 10 → 11 → 11 → 01 → 00. Reading the outputs, we see that the decoded sequence is {**111**, **100**, **011**, **100**, **111**}, and corresponds to the information sequence **1 1 1 0 0**.

P.2 ➡ *Solution*

Problem 2.1: The decoder diagram is sketched below.



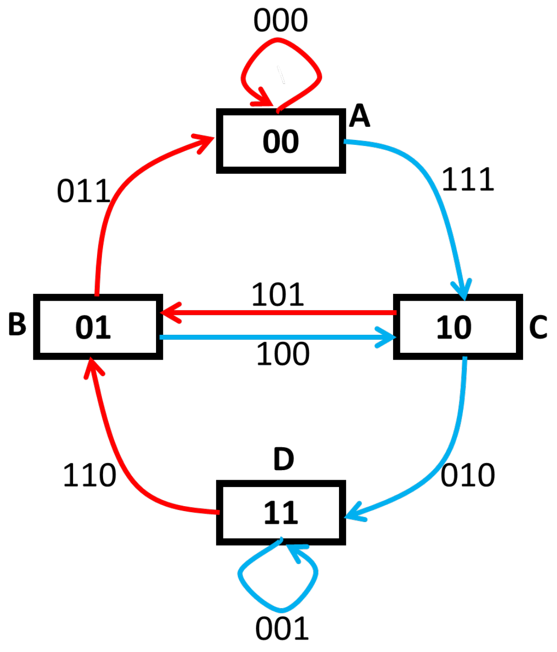
Problem 2.2: To draw the state-transition diagram, we first label states with four letters *A*, *B*, *C*, and *D*, as shown.

State	Label
00	*A*
01	*B*
10	*C*
11	*D*

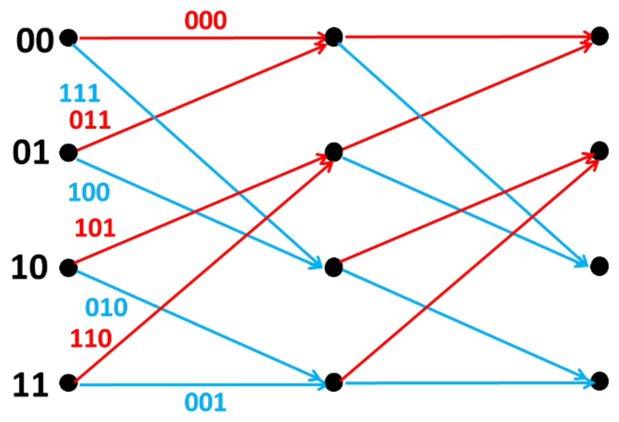
The code operations can be described by the equations $x_1 = m \oplus m_1$, $x_2 = m \oplus m_2$, and $x_3 = m_1 \oplus x_2$. We proceed to draw up the following table.

Input, m	m_1	m_2	Outputs x_1	x_2	x_3	Current state	Next state
0	0	0	0	0	0	*A*	*A* 000 → 000
1	0	0	1	1	1	*A*	*C* 100 → 010
0	0	1	0	1	1	*B*	*A* 001 → 000
1	0	1	1	0	0	*B*	*C* 101 → 010
0	1	0	1	0	1	*C*	*B* 010 → 001
1	1	0	0	1	0	*C*	*D* 110 → 011
0	1	1	1	1	0	*D*	*B* 011 → 001
1	1	1	0	0	1	*D*	*D* 111 → 011

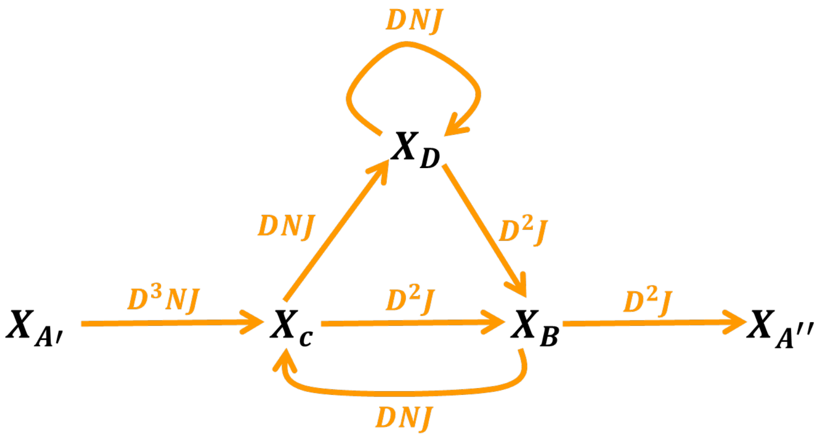
The state diagram is drawn next.



Problem 2.3: The trellis diagram is shown on the next page.



Problem 2.4: The transfer diagram is shown below.



The diagram above can be used to assemble the system of equations

$$X_B = D^2 J X_C + D^2 J X_D$$
$$X_C = D^3 NJ X_{A'} + DNJ X_B$$
$$X_D = DNJ X_C + DNJ X_D$$
$$X_{A''} = D^2 J X_{B\backslash}$$

Eliminating X_B, X_C, and X_D results in

$$T(D,N,J) = \frac{X_{A''}}{X_{A'}} = \frac{D^7 NJ^3}{1 - DNJ - D^3 NJ^2}$$

To find the free distance of the code we set $N = J = 1$ in the transfer function, giving

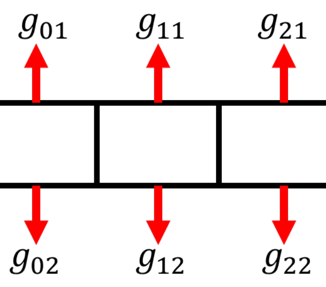
$$T_1(D) = T(D,N,J)|_{N=J=1} = \frac{D^7}{1 - D - D^3} = D^7 + D^8 + D^9 + \ldots$$

Clearly, $d_f = 7$.

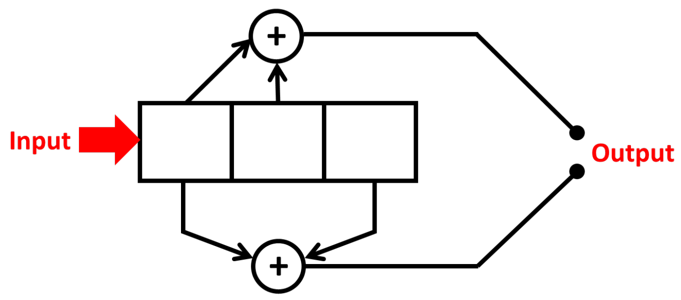
Problem 2.5: Since there is no self-loop corresponding to an input equal to 1 such that the output is the all-zero sequence, the code is not catastrophic.

P.3 ➡ *Solution*

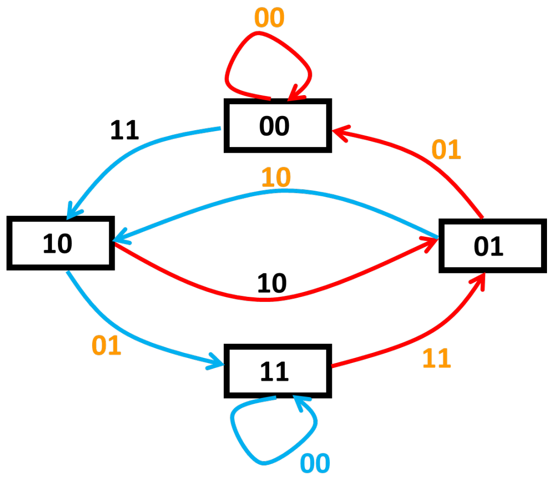
The hypothetical connections of the encoder are illustrated below. The code vectors are (g_{01}, g_{11}, g_{21}) and (g_{02}, g_{12}, g_{22}).



Note that as the system changes from 00 to 10, a branch word 11 is the output; this implies that $g_{01} = g_{02} = 1$. Similarly, as the system changes from 10 to 01, a branch word 10 is the output; it follows that $g_{11} = 1$ and $g_{12} = 0$. Finally, during a state transition from 11 to 11 itself, the branch word output is 00, which implies that $g_{21} = 0$ and $g_{22} = 1$. The code vectors are then (1,1,0) and (1,0,1), and the encoder is shown on the next page.



The complete state diagram is shown below.

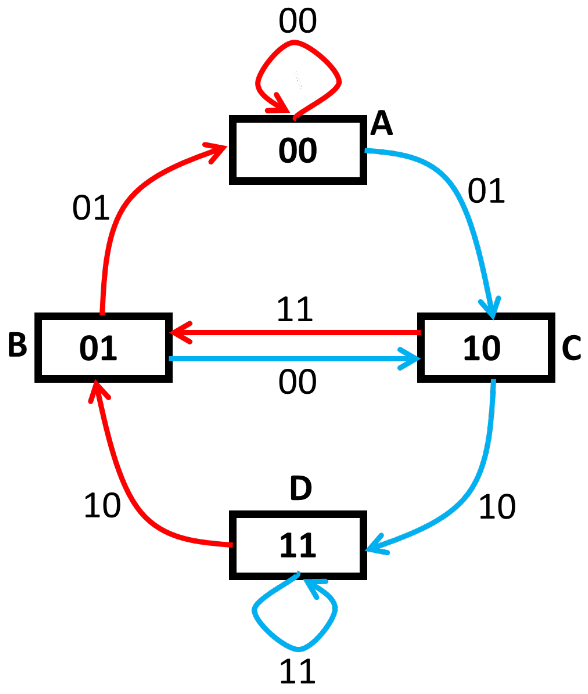


P.4 ➡ *Solution*

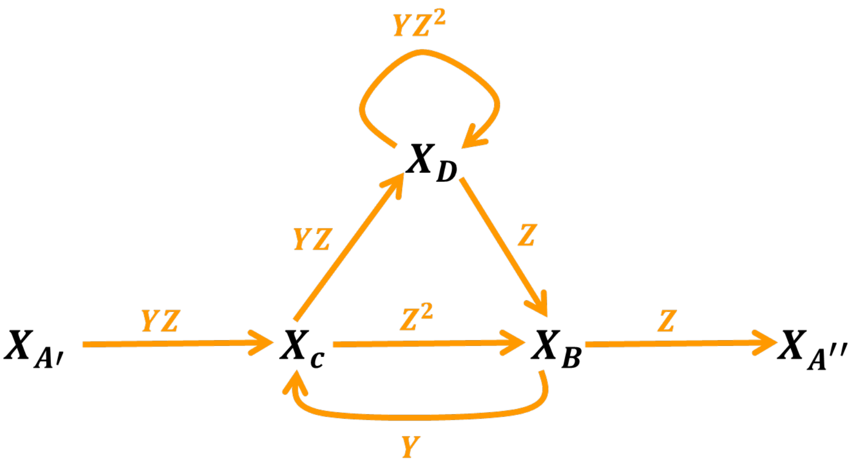
Problem 4.1: To tabulate the operation of the decoder, note that the outputs are described by the equations $x_1 = m_1$ and $x_2 = m \oplus m_1 \oplus m_2$.

Input, m	m_1	m_2	Outputs x_1	Outputs x_2	Current state	Next state
0	0	0	0	0	*A*	*A* 000 → 000
1	0	0	0	1	*A*	*C* 100 → 010
0	0	1	0	1	*B*	*A* 001 → 000
1	0	1	0	0	*B*	*C* 101 → 010
0	1	0	1	1	*C*	*B* 010 → 001
1	1	0	1	0	*C*	*D* 110 → 011
0	1	1	1	0	*D*	*B* 011 → 001
1	1	1	1	1	*D*	*D* 111 → 011

The state diagram is drawn next.



Problems 4.2 and 4.3: The transition diagram is shown below.



The pertaining equations are

$$\begin{aligned} X_C &= YZX_{A'} + YX_B \\ X_B &= Z^2 X_C + ZX_D \\ X_D &= YZX_C + YZ^2 X_D \\ X_{A''} &= ZX_B \end{aligned}$$

Eliminating X_B, X_C and X_D, we obtain

$$T(Y,Z) = \frac{X_{A''}}{X_{A'}} = \frac{YZ^4 + Y^2Z^4 - Y^2Z^6}{1 - 2YZ^2 - Y^2Z^2 + Y^2Z^4}$$

which can be expanded as

$$T(Y,Z) = YZ^4 + Y^2Z^4 + 3Y^3Z^6 + Y^4Z^6 + \ldots$$

Hence, $d_f = 4$.

Problems 4.4: Setting $Y = 1$, the transfer function becomes $T(Z) = 2Z^4 + 4Z^6 + \ldots$, hence $a_4 = 2$ and $a_6 = 4$. The bit error probability is bounded by

$$P_2(4) = \sum_{k=3}^{4} \binom{4}{k} p^k (1-p)^{4-k} + \frac{1}{2}\binom{4}{2} p^2 (1-p)^2 \approx 3p^2$$

Hence,

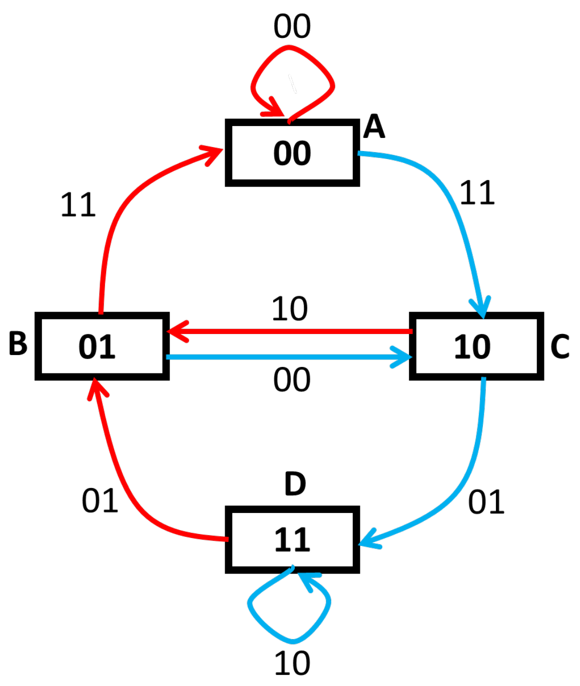
$$P_e \le a_4 \times P_2(4) = 2 \times \left[3 \times \left(10^{-3}\right)^2\right] = \boxed{6 \times 10^{-6}}$$

P.5 ➡ *Solution*

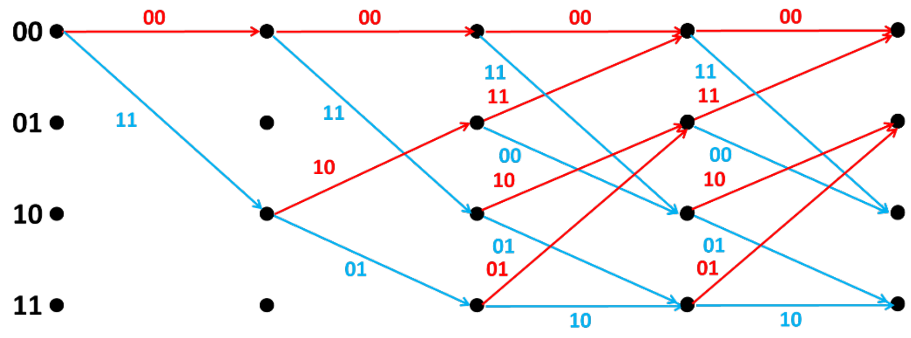
Problem 5.1: The output can be described by the operations $x_1 = m \oplus m_1 \oplus m_2$ and $x_2 = m \oplus m_2$. We proceed to draw up a state table.

Input, m	m_1	m_2	Outputs		Current state	Next state
			x_1	x_2		
0	0	0	0	0	*A*	*A* 000 → 000
1	0	0	1	1	*A*	*C* 100 → 010
0	0	1	1	1	*B*	*A* 001 → 000
1	0	1	0	0	*B*	*C* 101 → 010
0	1	0	1	0	*C*	*B* 010 → 001
1	1	0	0	1	*C*	*D* 110 → 011
0	1	1	0	1	*D*	*B* 011 → 001
1	1	1	1	0	*D*	*D* 111 → 011

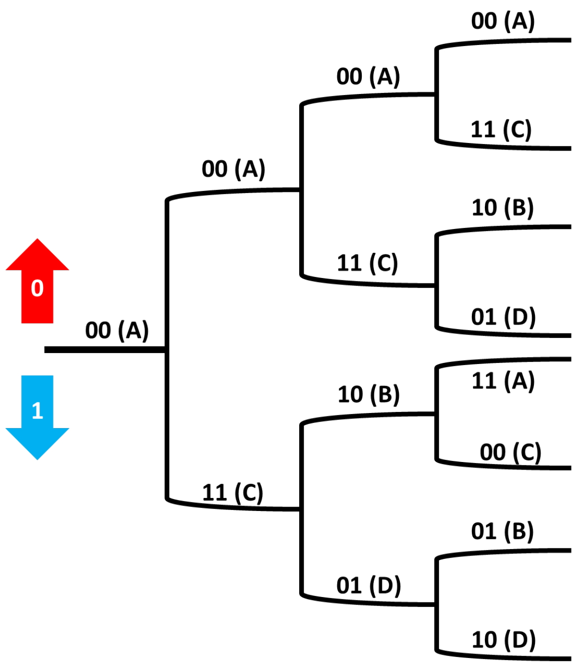
We proceed to draw the state diagram.



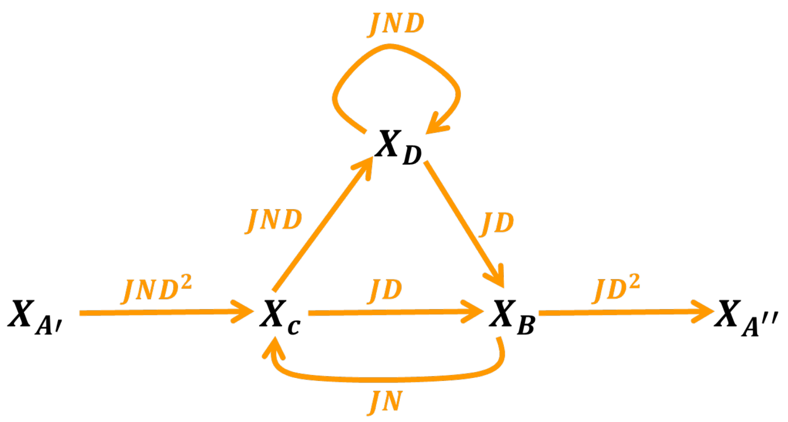
The trellis diagram is shown below.



The tree diagram is shown in continuation.



Problem 5.2: The transfer diagram is shown below.



Referring to the diagram above, we assemble the system of equations

$$X_C = JND^2 X_{A'} + JNX_B$$
$$X_B = JDX_C + JDX_D$$

$$X_D = JNDX_D + JNDX_C = NX_B$$

Expressing X_B in terms of $X_{A'}$, we write

$$X_B = \frac{J^2ND^3}{1-JND(1+J)}X_{A'}$$

However,

$$X_{A''} = JD^2X_B$$

so that

$$\frac{X_{A''}}{X_{A'}} = T(D,N,J) = \frac{JD^2 \times \dfrac{J^2ND^3\cancel{X_{A'}}}{1-JND(1+J)}}{\cancel{X_{A'}}} = \frac{J^3ND^5}{1-JND(1+J)}$$

This latter expression can be expanded as

$$T(D,N,J) = \frac{J^3ND^5}{1-JND(1+J)} = J^3ND^5 + J^4N^2D^6(1+J)+\ldots$$

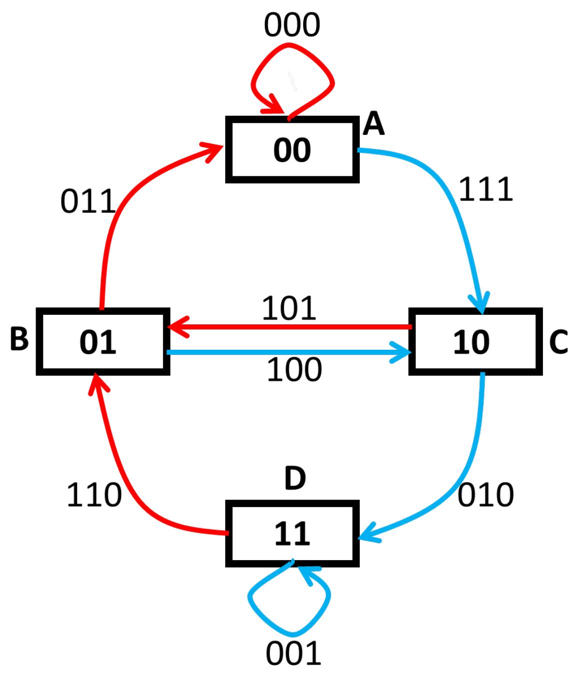
From the lowest exponent of D, we see that $d_{min} = 5$.

P.6 ➡ *Solution*

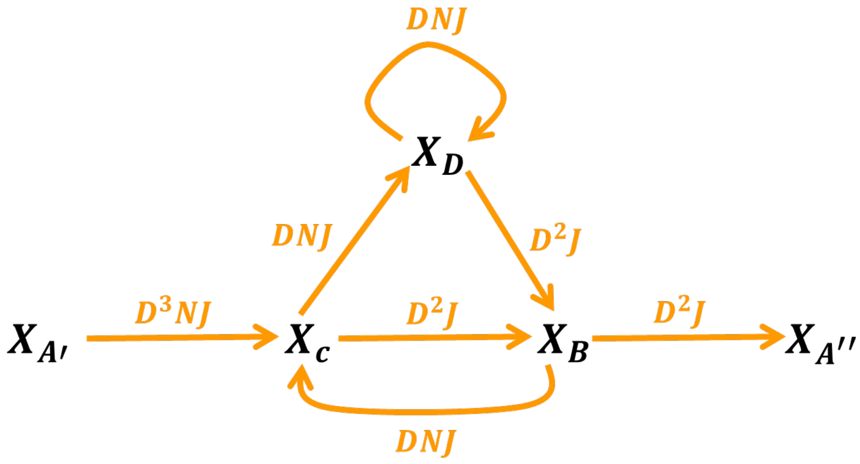
Problem 6.1: The code operations can be described by the equations $x_1 = m \oplus m_1$, $x_2 = m \oplus m_2$, and $x_3 = x_2 \oplus m_1$. The outputs and state transitions are tabulated below.

Input, m	m_1	m_2	Outputs			Current state	Next state
			x_1	x_2	x_3		
0	0	0	0	0	0	A	A 000 → 000
1	0	0	1	1	1	A	C 100 → 010
0	0	1	0	1	1	B	A 001 → 000
1	0	1	1	0	0	B	C 101 → 010
0	1	0	1	0	1	C	B 010 → 001
1	1	0	0	1	0	C	D 110 → 011
0	1	1	1	1	0	D	B 011 → 001
1	1	1	0	0	1	D	D 111 → 011

The state diagram is shown below.



Problems 6.2 and 6.3: To find the transfer function, we first sketch the transition diagram.



Then, we assemble the system of equations

$$X_C = D^3NJX_{A'} + DNJX_B$$
$$X_B = D^2JX_C + D^2JX_D$$
$$X_D = DNJX_C + DNJX_D$$
$$X_{A''} = D^2JX_B$$

which can be used to obtain the transfer function

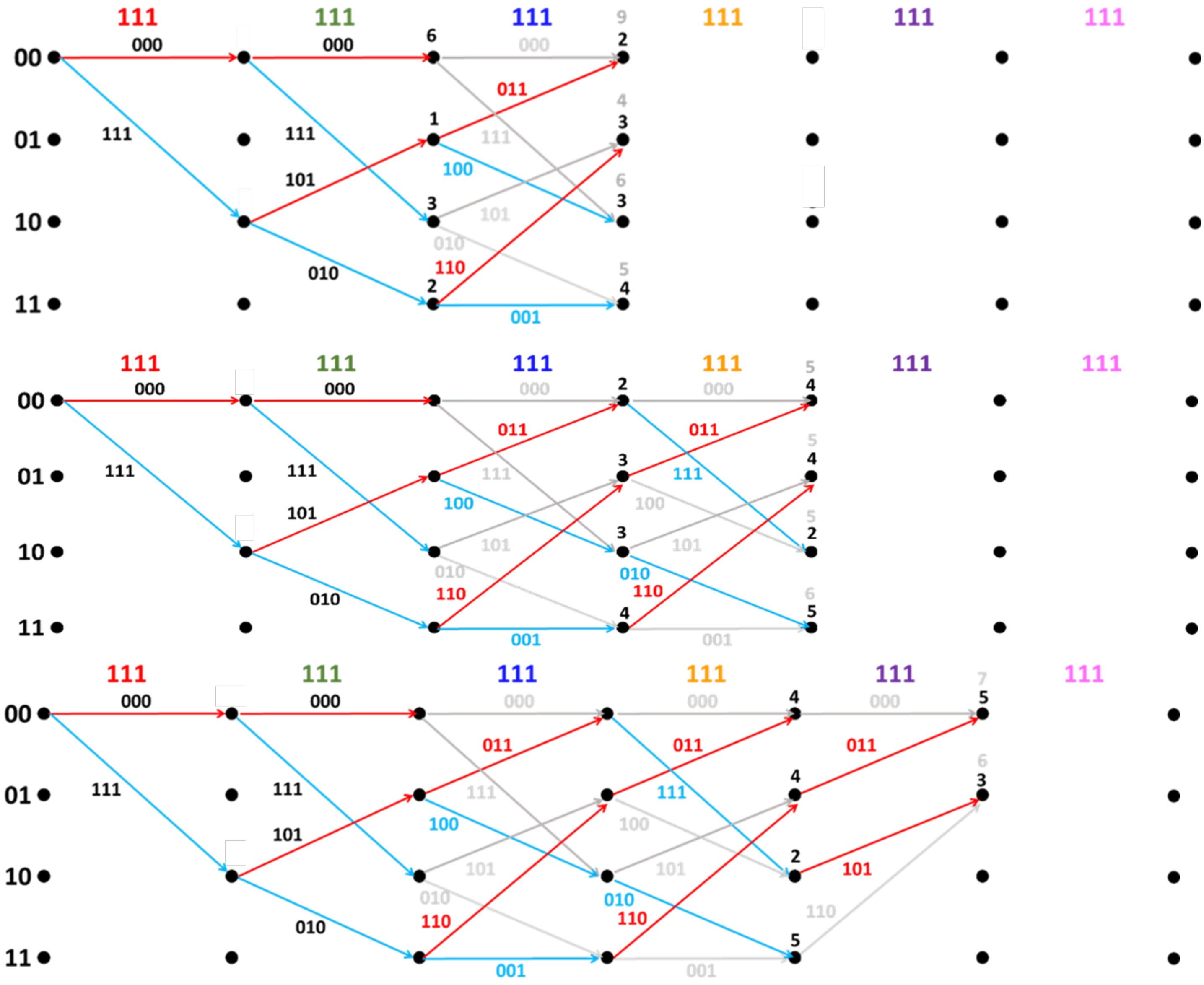
$$T(D,N,J) = \frac{X_{A''}}{X_{A'}} = \frac{D^7NJ^3}{1 - DNJ - D^3NJ^2}$$

To find the free distance of the code we set $N = J = 1$ in the transfer function, which leads to

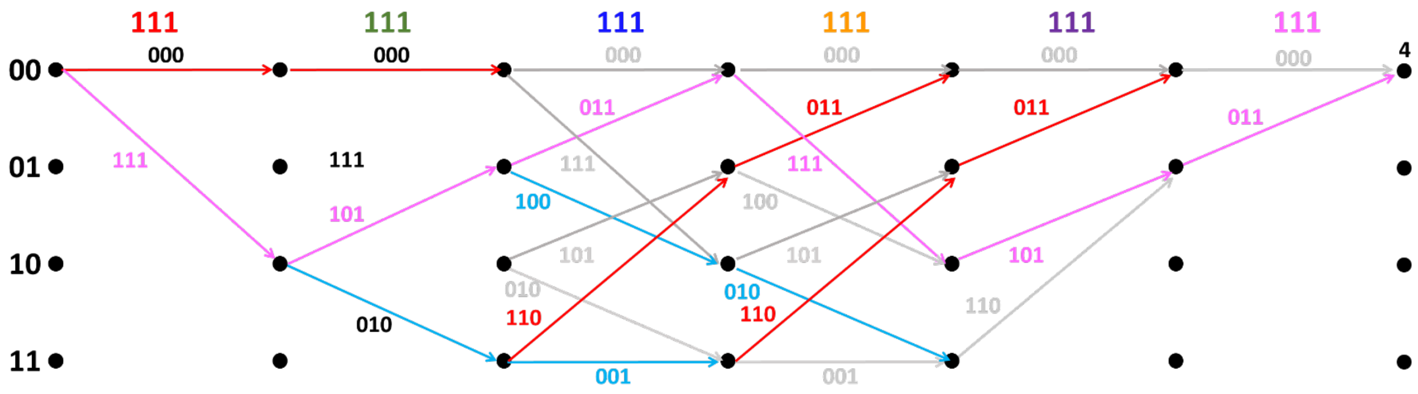
$$T_1(D) = T(D,N,J)\big|_{N=J=1} = \frac{D^7}{1 - D - D^3} = D^7 + D^8 + D^9 + 2D^{10} + \ldots$$

Therefore, $d_f = 7$.

Problem 6.4: The implementation of the Viterbi algorithm is no different from the approach adopted in Problem 1.6. The solution is developed below.



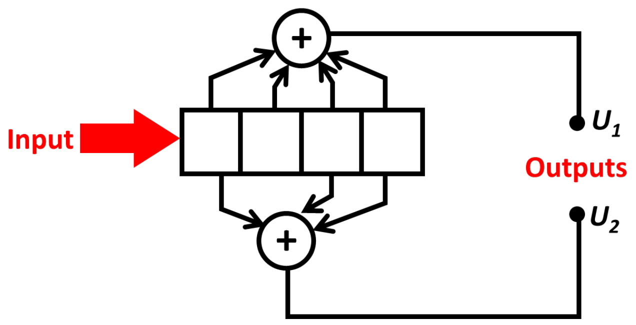
The surviving path is highlighted in pink.



Reading the outputs, we see that the decoded sequence is {**111**, **101**, **011**, **111**, **101**, **011**}, and corresponds to the information sequence **1 0 0 1 0 0**.

P.7 ➡ *Solution*

With reference to Table 1, we see that an optimum short constraint length convolutional code with constraint length K = 4 and rate 1/2 should have code vectors [1111] and [1011]. The corresponding decoder is illustrated below.



There are eight states, which we label as follows.

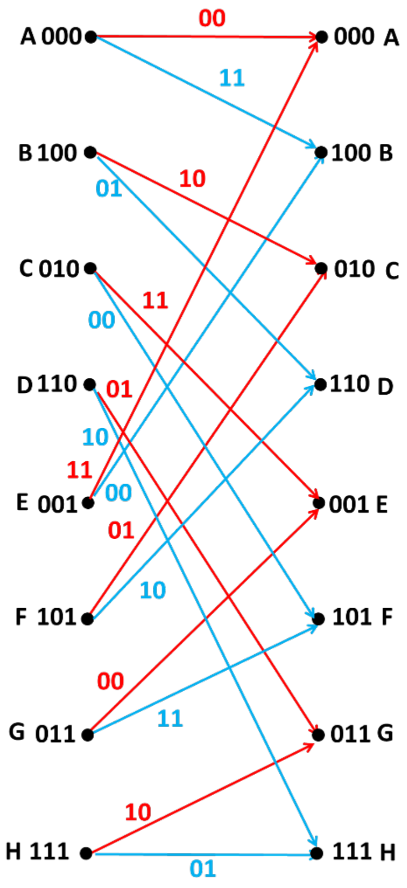
State	Label
000	A
100	B
010	C
110	D
001	E
101	F
011	G
111	H

We proceed to tabulate outputs $U_1 = m \oplus m_1 \oplus m_2 \oplus m_3$, $U_2 = m \oplus m_2 \oplus m_3$, and the corresponding state changes.

Input m	m_1	m_2	m_3	Outputs U_1	U_2	Curr. State	Next state
0	0	0	0	0	0	*A*	*A* 0000 → 0000
1	0	0	0	1	1	*A*	*B* 1000 → 0100
0	1	0	0	1	0	*B*	*C* 0100 → 0010
1	1	0	0	0	1	*B*	*D* 1100 → 0110
0	0	1	0	1	1	*C*	*E* 0010 → 0001
1	0	1	0	0	0	*C*	*F* 1010 → 0101
0	1	1	0	0	1	*D*	*G* 0110 → 0011
1	1	1	0	1	0	*D*	*H* 1110 → 0111
0	0	0	1	1	1	*E*	*A* 0001 → 0000
1	0	0	1	0	0	*E*	*B* 1001 → 0100

0	1	0	1	0	1	*F*	*C* 0101 → 0010
1	1	0	1	1	0	*F*	*D* 1101 → 0110
0	0	1	1	0	0	*G*	*E* 0011 → 0001
1	0	1	1	1	1	*G*	*F* 1011 → 0101
0	1	1	1	1	0	*H*	*G* 0111 → 0011
1	1	1	1	0	1	*H*	*H* 1111 → 0111

The trellis diagram is shown next.



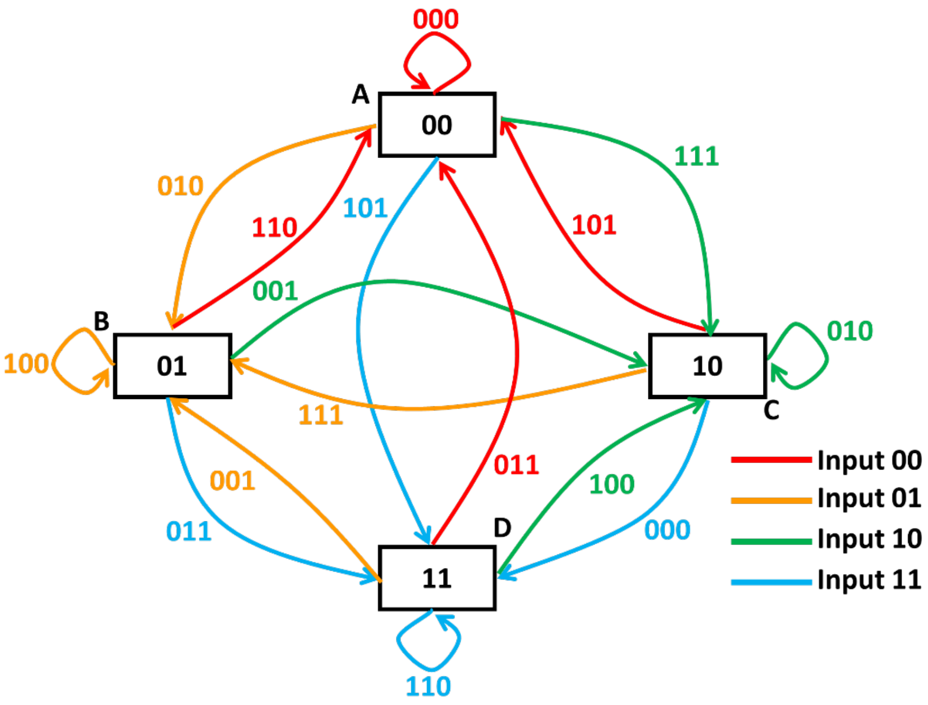
P.8 ➡ *Solution*

The bit positions in the register, from left to right, are labeled m_0, m_1, m_2, and m_3. The outputs are described by the equations $x_1 = m_0 \oplus m_2 \oplus m_3$, $x_2 = m_0 \oplus m_1 \oplus m_3$, and $x_3 = m_0 \oplus m_2$. As usual, the states are defined such that 00 = *A*, 01 = *B*, 10 = *C*, and 11 = *D*. The outputs and state transitions are tabulated below.

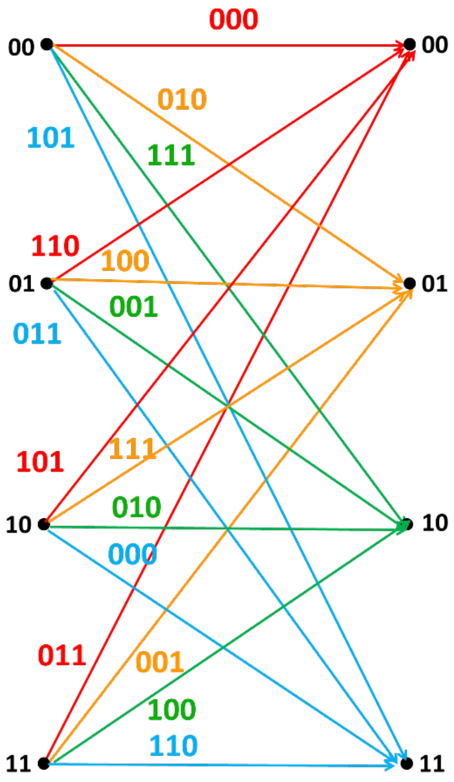
m_0	m_1	m_2	m_3	Outputs			Current state	Next state
				x_1	x_2	x_3		
0	0	0	0	0	0	0	*A*	*A* 0000 → 0000
0	1	0	0	0	1	0	*A*	*B* 0100 → 0001
1	0	0	0	1	1	1	*A*	*C* 1000 → 0010
1	1	0	0	1	0	1	*A*	*D* 1100 → 0011
0	0	0	1	1	1	0	*B*	*A* 0001 → 0000
0	1	0	1	1	0	0	*B*	*B* 0101 → 0001
1	0	0	1	0	0	1	*B*	*C* 1001 → 0010
1	1	0	1	0	1	1	*B*	*D* 1101 → 0011
0	0	1	0	1	0	1	*C*	*A* 0010 → 0000
0	1	1	0	1	1	1	*C*	*B*

								0110 → 0001
1	0	1	0	0	1	0	*C*	*C* 1010 → 0010
1	1	1	0	0	0	0	*C*	*D* 1110 → 0011
0	0	1	1	0	1	1	*D*	*A* 0011 → 0000
0	1	1	1	0	0	1	*D*	*B* 0111 → 0001
1	0	1	1	1	0	0	*D*	*C* 1011 → 0010
1	1	1	1	1	1	0	*D*	*D* 1111 → 0011

The state diagram is drawn next.



Equipped with the state diagram, we can easily sketch the trellis diagram.



REFERENCES

- PROAKIS, J.G. and SALEHI, M. (2008). *Digital Communications*. 5th edition. New York: McGraw-Hill.
- SKLAR, B. (2001). *Digital Communications*. 2nd edition. Upper Saddle River: Prentice-Hall.