Convolutional Codes

Paridhi Latawa

June 4, 2021

Abstract

This document covers convolutional codes. Reference texts include the following. Practice exercises are taken from Hankerson's texts.

1. Coding Theory and Cryptography: The Essentials, Second Edition by Hankerson
    a. Chapter 8: Convolutional Codes

2. MIT Course 6.02 Introduction to EECS II.
    a. Lecture 8: Convolutional Coding
    b. Lecture 9: Viterbi Decoding of Convolutional Codes

1 Coding Theory and Cryptography: The Essentials. Chapter 8: Convolutional Codes

Exercise 8.1.6 - a, d: Use the shift registers constructed in Exercise 8.1.5 to compute $a(x)g(x) = c(x)$. Compute $a(x)g(x)$ directly and compare the results.

a) $g(x) = 1 + x^2, a(x) = 1 + x$

If we were to take the input polynomial $a(x) = 1 + x$, such that we have values of $a(x)$ from $a_0, a_1, ..., a_6$, this polynomial corresponds to 1100000.

Assume the initial state of the 3 registers, given by the polynomial $g(x) = 1 + x^2$, is 000.

The output statement, with respect to mod 2, can be computed as 1111000 when creating a register and output table of the shifts.

This output sequence translates to the polynomial $c(x) = 1 + x + x^2 + x^3$.

$a(x)g(x)$ can also be computed directly.

$a(x)g(x) = (1 + x)(1 + x^2)$

$= 1 + x^2 + x + x^3$

$= 1 + x + x^2 + x^3$

$= c(x)$

d) $g(x) = 1 + x^3 + x^4, a(x) = x^2 + x^5 + x^6$

If we were to take the input polynomial $a(x) = x^2 + x^5 + x^6$, such that we have values of $a(x)$ from $a_0, a_1, ..., a_6$, this polynomial corresponds to 0010011. Note that the input polynomial does not have to be a set length and additional zeroes could be added after the last digit.

Assume the initial state of the 5 registers, given by the polynomial $g(x) = 1+x^3+x^4$, is 00000.

The output statement, with respect to mod 2, can be computed as 0010000 when creating a register and output table of the shifts.

This output sequence translates to the polynomial

$a(x)g(x)$ can also be computed directly.

$a(x)g(x) = (x^2 + x^5 + x^6)(1 + x^3 + x^4)$

$= x^{10} + 2x^9 + x^8 + 2x^6 + 2x^6 + x^2$

$= c(x)$

When simplifying $c(x)$ such that it follows modulus 2 and it is in the correct codomain space, we get the following as the output polynomial.

$c(x) = x^2 + x^{10}$

Note that quantities that had 2 as a coefficient were removed as they would simplify to a coefficient of 0 mod 2.

Note that terms that passed the $n-1$ power could be removed as these terms are not in the respective degree space of the shift register. They are not included in the shift register computed but are present in the multiplied polynomial.

The computed $c(x)$ matches the constructed output function from the shift register.

Exercise 8.1.7 - a, b: For the shift register in Figure 8.1, with $g(x) = 1 + x + x^3$, compute the output sequence $c_0, c_1, ...$ for each input sequence $a_0, a_1, ...$ below. Assume registers are all initially zero.

a) 10101000

We know that the generator polynomial for this shift register is $g(x) = 1 + x + x^3$

From the provided input sequence, we can devise the following input polynomial: $a(x) = 1 + x^2 + x^4$

There are two ways the output sequence can be computed: use the constructed shift register and table method, or compute the output polynomial $c(x)$ by multiplying $a(x)g(x)$. Both methods have been implemented in Exercise 8.1.6.

For sake of simplicity, let us use the second method to compute.

So,

$c(x) = a(x)g(x)$

$c(x) = (1 + x^2 + x^4)(1 + x + x^3)$

Multiplying this polynomial out, we get

$c(x) = 1 + x + x^2 + 2x^3 + x^4 + 2x^5 + x^7$

Simplifying out this polynomial based on modulus 2 and register degree constraints, the output polynomial comes out to be

$c(x) = 1 + x + x^2 + x^4 + x^7$

This corresponds to the following output sequence: 1110100

b) 0011000

We know that the generator polynomial for this shift register is $g(x) = 1 + x + x^3$

From the provided input sequence, we can devise the following input polynomial: $a(x) = x^2 + x^3$

There are two ways the output sequence can be computed: use the constructed shift register and table method, or compute the output polynomial $c(x)$ by multiplying $a(x)g(x)$. Both methods have been implemented in Exercise 8.1.6.

For sake of simplicity, let us use the second method to compute.

So,

$c(x) = a(x)g(x)$

$c(x) = (x^2 + x^3)(1 + x + x^3)$

Multiplying this polynomial out, we get

$c(x) = x^2 + 2x^3 + x^4 + x^5 + x^6$

Simplifying out this polynomial based on modulus 2 and register degree constraints, the output polynomial comes out to be

$c(x) = x^2 + x^4 + x^5 + x^6$

This corresponds to the following output sequence: 0010111

Exercise 8.1.12 - a: Given the feedback shift register in Figure 8.2, with the registers initially set to zero, generate the output sequence for each of the following received words. Indicate the final state of the registers and quotient if the remainder is zero.

a) 0011010

From Figure 8.2, we know that that the generator polynomial is $g(x) = 1 + x + x^3$

The received polynomial can be written as $c(X) = x^2 + x^3 + x^5$ as it corresponds to 0011010.

We can use the Feedback Shift Register (FSR) given in Figure 8.2 to create a table.

Time	Input	X0	X1	X2	Output
-1	—	0	0	0	—
0	1	0	0	0	0
1	0	0	1	0	0
2	1	1	0	1	0
3	1	1 + 1	1 + 1	0	1
4	0	0	0	0	0
5	0	0	0	0	0

$\frac{c(x)}{g(x)} = x^2$. This is the quotient and corresponds to the output sequence in the reverse order.

The remainder is also 000.

Note that the input was also inputted into the table in reverse order.

Exercise 8.2.2 - a, b: Encode the following messages using the $(3, 1, 3)$ convolutional code with generators $g_1(x) = 1 + x + x^3, g_2(x) = 1 + x + x^2 + x^3$, and $g_3(x) = 1 + x^2 + x^3$.

a) $m(x) = 1 + x^3$

Unpacking the initial definition of the convolutional code, we know that a : Encode the following messages using the (3, 2, 4) convolutional code with generators $g_1(x) = 1 + x^3, g_2(x) = x + x^4$, and $g_3(x)1 + x + x^2 + x^3 + x^4$. Use both techniques of encoding described above.

In this case, we have a $(3, 1, 3)$ convolutional code. We see that it has $n = 3$ generators.

In general, a message $m(x)$, where $m(x) = m_0 + m_1x + m_2x^2 + ... \in K[x]$, with generators $g_1(x), ..., g_n(x)$ such that $g_i(x) = g_{i,0} + g_{i,1}x + ... + g_{i,m}x^m, g_i \in K[x]$ can be encoded to the codeword $c(x)$ as $c_i(x) = m(x)g_i(x)$.

Based on the above general definition, the given message $m(x) = 1 + x^3$ can be encoded as the following:

$c(x) = ((1 + x^3)g_1(x), (1 + x^3)g_2(x), (1 + x^3)g_3(x))$

$= ((1 + x^3)(1 + x + x^3), (1 + x^3)(1 + x + x^2 + x^3), (1 + x^3)(1 + x + x^2 + x^3))$

$= (1 + x + 2x^3 + x^4 + x^6, 1 + x + x^2 + 2x^3 + x^4 + x^5 + x^6, 1 + x^2 + 2x^3 + x^5 + x^6)$

$= (1 + x + x^4 + x^6, 1 + x + x^2 + x^4 + x^5 + x^6, 1 + x^2 + x^5 + x^6)$

$\longleftrightarrow (1100101..., 1110111..., 1010011...)$

b) $m(x) = 1 + x + x^3$

Based on the definition provided in 8.2.2a, the given message $m(x) = 1 + x + x^3$ can be encoded as the following:

$c(x) = ((1 + x + x^3)g_1(x), (1 + x + x^3)g_2(x), (1 + x + x^3)g_3(x)) = ((1 + x + x^3)(1 + x + x^3), (1 + x + x^3)(1 + x + x^2 + x^3), (1 + x + x^3)(1 + x + x^2 + x^3))$

$= (1+2x+x^2+2x^3, 1+2x+2x^2+3x^3+2x^4+x^5+x^6, 1+x+x^2+3x^3+x^4+x^5+x^6)$

$= (1 + x^2 + x^6, 1 + x^3 + x^5 + x^6, 1 + x + x^2 + x^3 + x^4 + x^5 + x^6)$

$\longleftrightarrow (1010001..., 1001011..., 1111111...)$

Exercise 8.2.3 - a, c: Encode the following messages using the (2, 1, 4) convolutional code with generators $g_1(x) = 1 + x^3 + x^4$ and $g_2(x) = 1 + x + x^2 + x^4$

a) $m(x) = 1 + x + x^2$

Based on the definition provided in 8.2.2a, the given message $m(x) = 1 + x + x^2$ can be encoded as the following:

$c(x) = ((1 + x + x^2)g_1(x), (1 + x + x^2)g_2(x))$

$= ((1 + x + x^2)(1 + x^3 + x^4), (1 + x + x^2)(1 + x + x^2 + x^4))$

$= (1 + x + x^2 + x^3 + 2x^4 + 2x^5 + x^6, 1 + 2x + 3x^2 + 2x^3 + 2x^4 + x^5 + x^6)$

$= (1 + x + x^2 + x^3 + x^6, 1 + x^2 + x^5 + x^6)$

$\longleftrightarrow (1111001..., 1010011...)$

c) $m(x) = 1 + x^2 + x^4 + ... = \sum_{i=0}^{\infty} x^{2i}$

Based on the definition provided in 8.2.2a, the given message $m(x) = 1 + x^2 + x^4 + ... = \sum_{i=0}^{\infty} x^{2i}$ can be encoded as the following:

$c(x) = (1 + x^2 + x^4 + ... = \sum_{i=0}^{\infty} x^{2i})(g_1(x)), (1 + x^2 + x^4 + ... = \sum_{i=0}^{\infty} x^{2i})(g_2(x))$

$= (1+x^2+x^3+2x^4+x^5+x^6+x^7+x^8+...., 1+x+2x^2+x^3+3x^4+x^5+2x^6+x^8+....)$

$= (1 + x^2 + x^3 + x^5 + x^6 + x^7 + x^8 + ..., 1 + x + x^3 + x^4 + x^5 + x^8 + ...)$

$= (1 + x^2 + \sum_{i=3}^{\infty} x^{2i}, 1 + x + \sum_{i=3}^{\infty} x^{2i})$

Exercise 8.2.6: For the convolutional codes in Exercises 8.2.2 and 8.2.3, construct the relevant shift register which can be used to encode the code. Then check your answers to those exercises by using the shift register to encode the given message. Finally, represent each codeword in its interleaved form.

Exercise 8.2.2a provides the following action item: Encode the message $m(x) = 1 + x^3$ message using the $(3, 1, 3)$ convolutional code with generators $g_1(x) = 1 + x + x^3, g_2(x) = 1 + x + x^2 + x^3$, and $g_3(x) = 1 + x^2 + x^3$.

The message $m(x) = 1 + x^3$ corresponds to the input polynomial sequence 1001...

If we were to encode the message $m(x)$ using the above shift register, we can create a register-output table as shown below.

				Output	
Time	Input	$X_0X_1X_2X_3$	c_1	c_2	c_3
-1	—	0 0 0 0		—	
0	1	1 0 0 0	1	1	1
1	0	0 1 0 0	1	1	0
2	0	0 0 1 0	0	1	1
3	1	1 0 0 1	1	0	0

Thus, we get the interleaved output sequence

111 110 011 100 ...

In 8.2.2a, we computed the following to be the $c(X)$ output polynomial.

$c(x) = (1 + x + x^4 + x^6, 1 + x + x^2 + x^4 + x^5 + x^6, 1 + x^2 + x^5 + x^6)$

$\longleftrightarrow (1100101..., 1110111..., 1010011...)$

When obtaining the individual interleaved sequence, we get 111110011100, which corresponds to the above obtained interleaved output sequence from the shift register.

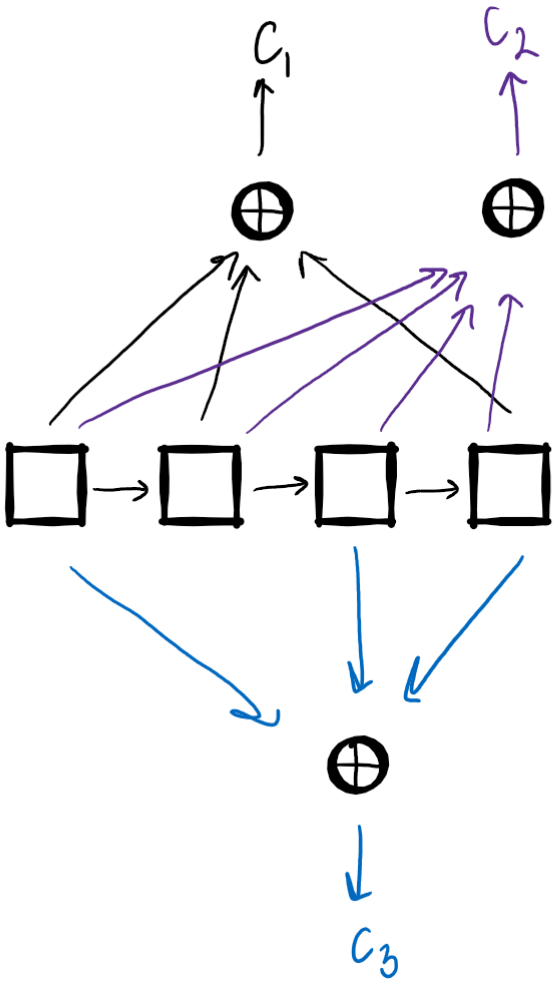

Figure 1: This is an image of the shift register for question 8.2.6

Exercise 8.2.8 - a: Encode the following messages using the (3, 2, 4) convolutional code with generators $g_1(x) = 1 + x^3, g_2(x) = x + x^4$, and $g_3(x)1 + x + x^2 + x^3 + x^4$. Use both techniques of encoding described above.

a) $m(x)1 + x + x^3 + x^4 + x^5$

The message polynomial corresponds to the message sequence $m = 110111...$

The first interpretation of $k = 2$ is to encode m using the single shift register shown below and moving $k = 2$ message digits in to the shift register at each tick.

The contents of the registers and the outputs are summarized in the following table.

				Output	
Time	Input	$X_0X_1X_2X_3X_4X_5$	c_1	c_2	c_3
-1	—	0 0 0 0 0 0		—	
0	11	1 1 0 0 0 0	1	1	0
1	01	0 1 1 1 0 0	1	1	1
2	11	1 1 0 1 1 1	0	0	0
3	00	0 0 1 1 0 1	1	0	0
4	00	0 0 0 0 1 1	0	1	1
5	00	0 0 0 0 0 0	0	0	0

So, m is encoded to the codeword (in interleaved form)

110 111 000 100 011 000...

The second interpretation of $k = 2$ is to notice that the first, third, fifth, ... message digits fed into the shift register only ever appear in X_0, X_2, and X_4, and the second, fourth, sixth, ... message digits fed into the shift register only appear in X_1 and X_3. Thus, the message and the registers can be split into $k = 2$ parts, as shown in the below figure.

Exercise 8.2.11

a) Find the state diagram, and its representation in tabular form, for the $(2, 1, 2)$ convolutional code with generators $g_1(x) = 1 + x^2$ and $g_2(x) = 1 + x + x^2$

There are four possible states for a binary sequence of length 2: 00, 01, 10, 11.

These are represented as four vertices on the state diagram.

b) Use the state diagram to encode the following messages

(i) $m(x) = 1 + x^2$

This polynomial corresponds to the sequence 101...

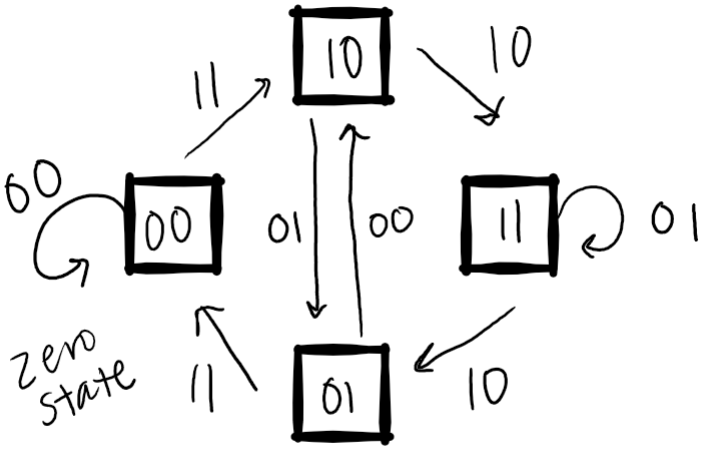

Figure 2: This is an image of the shift register for question 8.2.6

So, we get a directed walk giving the following outputs 00, 11, 01, 00. SO the corresponding encoded message would be 11 01 00 ..

(ii) $m(x) = 1 + x + x^2$

This polynomial corresponds to the sequence 111..

So, we get a directed walk giving the following outputs 00, 11, 10, 01, ...

SO the corresponding encoded message would be 11 10 01 ..

c) Use the state diagram to find the message corresponding to the following codewords

(i) 11 01 00 01 11 00...

The first digit of each state as we follow the directed path of the output will give the message.

So, we get the following digits for the message: 1, 0, 1, 0, 0, 0, corresponding to 101000.... as the input message.

(ii) 00 11 10 01 01 10 00...

The first digit of each state as we follow the directed path of the output will give the message.

So, we get the following digits for the message: 0, 1, 1, 1, 1, 0, 1 corresponding to 0111101 .... as the input message.

Exercise 8.3.3 - a, b

Exercise 8.3.6 - a

Exercise 8.4.5: Again, using the convolutional code C_1 with $g_1(x) = 1 + x + x$

Exercise 8.4.10

Exercise 8.4.14

References

References

Adleman Adleman, L. "Molecular Computation of Solutions to Combinatorial Problems." Science, vol. 266, no. 5187, 1994, pp. 1021–1024., doi:10.1126/science.7973651.Balakrishnan, Hari, and Jacob K. White. "ECTURE Convolutional Coding - MIT." Battail, Gérard. "Information Theory and Error-Correcting Codes In Genetics and Biological Evolution." Introduction to Biosemiotics, pp. 299–345., doi:10.1007/1-4020-4814-913 Elias, P. Coding for noisy channels. IRE Conv. Rec. 1955, 4, 37–46. Hankerson, D. R. et al. Coding Theory and Cryptography: The Essentials, 2nd ed, Marcel Dekker, 2000. Viraktamath, S V, director. How to Draw Trellis Diagram for a given Encoder. YouTube, YouTube, 1 Nov. 2018,