An introduction to coding theory

Adrish Banerjee

Department of Electrical Engineering
Indian Institute of Technology Kanpur
Kanpur, Uttar Pradesh
India

Feb. 13, 2017

Lecture #8A: Introduction to convolutional codes-I: Encoding

Convolutional codes

Outline of the lecture:

- Introduction

Convolutional codes

Outline of the lecture:

- Introduction
- Encoding for convolutional code

Introduction

- A convolutional encoder processes the information sequence continuously.

Introduction

- A convolutional encoder processes the information sequence continuously.
- The n-bit encoder output at a particular time depends not only on the k-bit information sequence, but also on m previous input blocks, i.e., a convolutional encoder has a memory order of m.

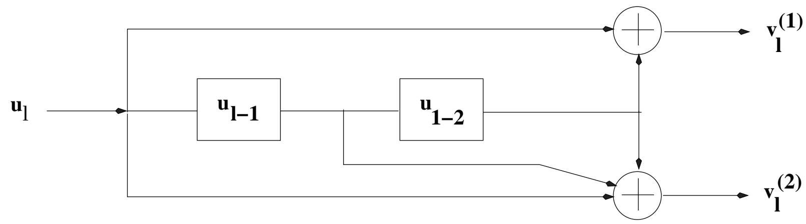
Introduction

- A convolutional encoder processes the information sequence continuously.
- The n-bit encoder output at a particular time depends not only on the k-bit information sequence, but also on m previous input blocks, i.e., a convolutional encoder has a memory order of m.
- The set of sequences produced by a k-input, n-output encoder of memory order m is called an (n, k, m) convolutional code.

Introduction

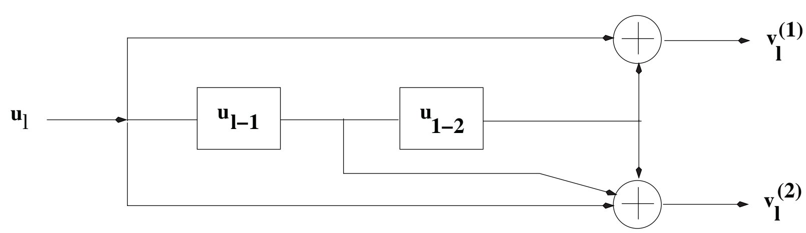
- A convolutional encoder processes the information sequence continuously.
- The n-bit encoder output at a particular time depends not only on the k-bit information sequence, but also on m previous input blocks, i.e., a convolutional encoder has a memory order of m.
- The set of sequences produced by a k-input, n-output encoder of memory order m is called an (n, k, m) convolutional code.
- The values of n and k are much smaller for convolutional codes compared to the block codes.

Introduction



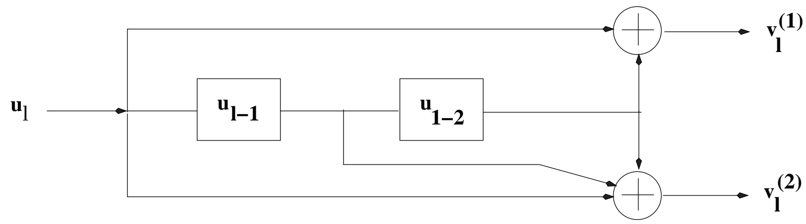
- Let $k = 1$, $n = 2$ and $m = 2$. The following circuit generates a $(2, 1, 2)$ convolutional code.

Introduction



- Let $k = 1$, $n = 2$ and $m = 2$. The following circuit generates a $(2, 1, 2)$ convolutional code.
- Input: u_l

Introduction



- Let $k = 1$, $n = 2$ and $m = 2$. The following circuit generates a $(2, 1, 2)$ convolutional code.
- Input: u_l
- Outputs:

$$\begin{aligned} v_l^{(1)} &= u_l + u_{l-2} \\ v_l^{(2)} &= u_l + u_{l-1} + u_{l-2} \end{aligned}$$

Introduction

- If ν_i is the length of the i^{th} shift register in a convolutional encoder with k input sequences, $i = 1, 2, \cdots, k$, then

$$m = \max_{1 \le i \le k} \nu_i$$

Introduction

- If ν_i is the length of the i^{th} shift register in a convolutional encoder with k input sequences, $i = 1, 2, \cdots, k$, then

$$m = \max_{1 \leq i \leq k} \nu_i$$

- The parameter m is known as memory order of the code.

Introduction

- If ν_i is the length of the i^{th} shift register in a convolutional encoder with k input sequences, $i = 1, 2, \cdots, k$, then

$$m = \max_{1 \leq i \leq k} \nu_i$$

- The parameter m is known as memory order of the code.
- The ratio $R = k/n$ is known as the code rate.

Introduction

- If ν_i is the length of the i^{th} shift register in a convolutional encoder with k input sequences, $i = 1, 2, \cdots, k$, then

$$m = \max_{1 \leq i \leq k} \nu_i$$

- The parameter m is known as memory order of the code.
- The ratio $R = k/n$ is known as the code rate.
- The overall constraint length ν of the encoder is defined as

$$\nu = \sum_{1 \leq i \leq k} \nu_i$$

Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence

$$\mathbf{u} = (u_0, u_1, \cdots, u_l, \cdots)$$

is encoded into n output sequences.

$$\begin{aligned}
\mathbf{v}^{(1)} &= (v_0^{(1)}, v_1^{(1)}, \cdots, v_l^{(1)}, \cdots) \\
\mathbf{v}^{(2)} &= (v_0^{(2)}, v_1^{(2)}, \cdots, v_l^{(2)}, \cdots) \\
&\vdots \\
\mathbf{v}^{(n)} &= (v_0^{(n)}, v_1^{(n)}, \cdots, v_l^{(n)}, \cdots)
\end{aligned}$$

Encoding of $(n, 1, m)$ convolutional code

- In this case, a single information sequence

$$\mathbf{u} = (u_0, u_1, \cdots, u_l, \cdots)$$

is encoded into n output sequences.

$$\begin{aligned}
\mathbf{v}^{(1)} &= (v_0^{(1)}, v_1^{(1)}, \cdots, v_l^{(1)}, \cdots) \\
\mathbf{v}^{(2)} &= (v_0^{(2)}, v_1^{(2)}, \cdots, v_l^{(2)}, \cdots) \\
&\vdots \\
\mathbf{v}^{(n)} &= (v_0^{(n)}, v_1^{(n)}, \cdots, v_l^{(n)}, \cdots)
\end{aligned}$$

- The n output sequences are interleaved to form a single code sequence.

$$\mathbf{v} = (\mathbf{v}_0, \mathbf{v}_1, \cdots, \mathbf{v}_l, \cdots), \text{ where } \mathbf{v}_l = (v_l^{(1)}, v_l^{(2)}, \cdots, v_l^{(n)})$$

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m+1$,

$$\begin{aligned}
\mathbf{g}^{(1)} &= (g_0^{(1)}, g_1^{(1)}, \cdots, g_m^{(1)}) \\
\mathbf{g}^{(2)} &= (g_0^{(2)}, g_1^{(2)}, \cdots, g_m^{(2)}) \\
&\vdots \\
\mathbf{g}^{(n)} &= (g_0^{(n)}, g_1^{(n)}, \cdots, g_m^{(n)})
\end{aligned}$$

Encoding of $(n, 1, m)$ convolutional code

- The code is specified by a set of n generator sequences of length $m+1$,

$$\begin{aligned}\mathbf{g}^{(1)} &= (g_0^{(1)}, g_1^{(1)}, \cdots, g_m^{(1)})\\ \mathbf{g}^{(2)} &= (g_0^{(2)}, g_1^{(2)}, \cdots, g_m^{(2)})\\ &\vdots\\ \mathbf{g}^{(n)} &= (g_0^{(n)}, g_1^{(n)}, \cdots, g_m^{(n)})\end{aligned}$$

- The output sequence is the discrete convolution of the information sequence $\mathbf{u}$ and the generator sequence $\mathbf{g}^{(i)}$, i.e.

$$\mathbf{v}^{(i)} = \mathbf{u} \star \mathbf{g}^{(i)}, \quad 1 \leq i \leq n,$$

and

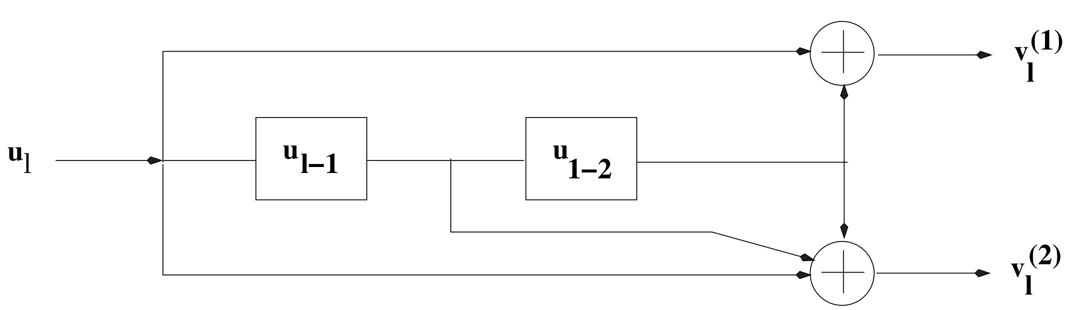
$$\begin{aligned}\mathbf{v}_l^{(i)} &= u_l g_0^{(i)} + u_{l-1} g_1^{(i)} + \cdots + u_{l-m} g_m^{(i)}\\ &= \sum_{j=0}^{m} u_{l-j} g_j^{(i)}\end{aligned}$$

Encoding of $(n, 1, m)$ convolutional code

Example:

- Consider a rate $R = 1/2$, $(2, 1, 2)$ convolutional code specified by the following generator sequences

$$\begin{aligned}\mathbf{g}^{(1)} &= (1\ 0\ 1),\\ \mathbf{g}^{(2)} &= (1\ 1\ 1),\end{aligned}$$



Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \cdots)$.

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \cdots)$.
- The output sequences are given by

$$\begin{aligned}\mathbf{v}^{(1)} &= (1\ 0\ 0\ 1\ 0\ 1\ 1\ \cdots)\\ \mathbf{v}^{(2)} &= (1\ 1\ 0\ 0\ 1\ 0\ 1\ \cdots)\end{aligned}$$

Encoding of $(n, 1, m)$ convolutional code

- Let the information sequence $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \cdots)$.
- The output sequences are given by

$$\begin{aligned} \mathbf{v}^{(1)} &= (1\ 0\ 0\ 1\ 0\ 1\ 1\ \cdots) \\ \mathbf{v}^{(2)} &= (1\ 1\ 0\ 0\ 1\ 0\ 1\ \cdots) \end{aligned}$$

- The code sequence can be written as

$$\mathbf{v} = (11,\ 01,\ 00,\ 10,\ 01,\ 10,\ 11,\ \cdots)$$

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix

- Matrix form

$$\mathbf{v} = \mathbf{u}\mathbf{G}$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_0 & \mathbf{g}_1 & \cdots & \cdots & \mathbf{g}_m & & \\ & \mathbf{g}_0 & \mathbf{g}_1 & \cdots & \cdots & \mathbf{g}_m & \\ & & \ddots & & & & \ddots \end{bmatrix}$$

where

$$\mathbf{g}_i = (g_i^{(1)}\ g_i^{(1)}\ \cdots\ g_i^{(n)}),\ \ 0 \le i \le m$$

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1\ 0\ 1)$, and $\mathbf{g}^{(2)} = (1\ 1\ 1)$, the generator matrix is given by

$$\mathbf{G} = \begin{bmatrix} 1\,1 & 0\,1 & 1\,1 & & & & \\ & 1\,1 & 0\,1 & 1\,1 & & & \\ & & 1\,1 & 0\,1 & 1\,1 & & \\ & & & 1\,1 & 0\,1 & 1\,1 & \\ & & & & 1\,1 & 0\,1 & 1\,1 \\ & & & & & \ddots & & \ddots \end{bmatrix}$$

Encoding of $(n, 1, m)$ convolutional code

Representation of encoding matrix: Example

- For $(2, 1, 2)$ convolutional code with $\mathbf{g}^{(1)} = (1\ 0\ 1)$, and $\mathbf{g}^{(2)} = (1\ 1\ 1)$, the generator matrix is given by

$$\mathbf{G} = \begin{bmatrix} 1\,1 & 0\,1 & 1\,1 & & & & \\ & 1\,1 & 0\,1 & 1\,1 & & & \\ & & 1\,1 & 0\,1 & 1\,1 & & \\ & & & 1\,1 & 0\,1 & 1\,1 & \\ & & & & 1\,1 & 0\,1 & 1\,1 \\ & & & & & \ddots & & \ddots \end{bmatrix}$$

- For $\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \cdots)$,

$$\mathbf{v} = \mathbf{uG} = (11,\ 01,\ 00,\ 10,\ 01,\ 10,\ 11,\ 00,\ 00,\ \cdots)$$

Encoding of $(n, 1, m)$ convolutional code

- Polynomial representation:

$$\begin{aligned} \mathbf{v}^{(i)}(D) &= \mathbf{u}(D)\mathbf{g}^{(i)}(D), \ ,1 \le i \le n \\ \mathbf{v}(D) &= \mathbf{v}^{(1)}(D^n) + D\mathbf{v}^{(2)}(D^n) + \cdots + D^{n-1}\mathbf{v}^{(n)}(D^n) \end{aligned}$$

Time Domain	Transform Domain
$\mathbf{g}^{(1)} = (1\ 0\ 1)$	$\mathbf{g}^{(1)}(D) = 1 + D^2$
$\mathbf{g}^{(2)} = (1\ 1\ 1)$	$\mathbf{g}^{(2)}(D) = 1 + D + D^2$
$\mathbf{u} = (1\ 0\ 1\ 1\ 1\ 0\ 0\ \cdots)$	$\mathbf{u}(D) = 1 + D^2 + D^3 + D^4$
$\mathbf{v}^{(1)} = \mathbf{u} \star \mathbf{g}^{(1)}$	$\mathbf{v}^{(1)}(D) = \mathbf{u}(D)\mathbf{g}^{(1)}(D)$
$= (1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ \cdots)$	$= 1 + D^3 + D^5 + D^6$
$\mathbf{v}^{(2)} = \mathbf{u} \star \mathbf{g}^{(2)}$	$\mathbf{v}^{(2)}(D) = \mathbf{u}(D)\mathbf{g}^{(2)}(D)$
$= (1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ \cdots)$	$= 1 + D + D^4 + D^6$

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,

$$\mathbf{v}(D) = \mathbf{u}(D^n)\mathbf{g}(D)$$

where

$$\mathbf{g}(D) \triangleq \mathbf{g}^{(1)}(D^n) + D\mathbf{g}^{(2)}(D^n) + \cdots + D^{n-1}\mathbf{g}^{(n)}(D^n)$$

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,

$$\mathbf{v}(D) = \mathbf{u}(D^n)\mathbf{g}(D)$$

where

$$\mathbf{g}(D) \triangleq \mathbf{g}^{(1)}(D^n) + D\mathbf{g}^{(2)}(D^n) + \cdots + D^{n-1}\mathbf{g}^{(n)}(D^n)$$

- Example: Time Domain

$$\mathbf{v} = \mathbf{uG} = (11,\ 01,\ 00,\ 10,\ 01,\ 10,\ 11,\ 00,\ 00,\ \cdots)$$

Encoding of $(n, 1, m)$ convolutional code

- The encoding equations can alternately written as,

$$\mathbf{v}(D) = \mathbf{u}(D^n)\mathbf{g}(D)$$

where

$$\mathbf{g}(D) \triangleq \mathbf{g}^{(1)}(D^n) + D\mathbf{g}^{(2)}(D^n) + \cdots + D^{n-1}\mathbf{g}^{(n)}(D^n)$$

- Example: Time Domain

$$\mathbf{v} = \mathbf{uG} = (11,\ 01,\ 00,\ 10,\ 01,\ 10,\ 11,\ 00,\ 00,\ \cdots)$$

- Example: Transform Domain

$$\begin{aligned} \mathbf{v}(D) &= \mathbf{u}(D^2)\mathbf{g}(D) = (1 + D^4 + D^6 + D^8)(1 + D + D^3 + D^4 + D^5) \\ &= 1 + D + D^3 + D^6 + D^9 + D^{10} + D^{12} + D^{13} \end{aligned}$$