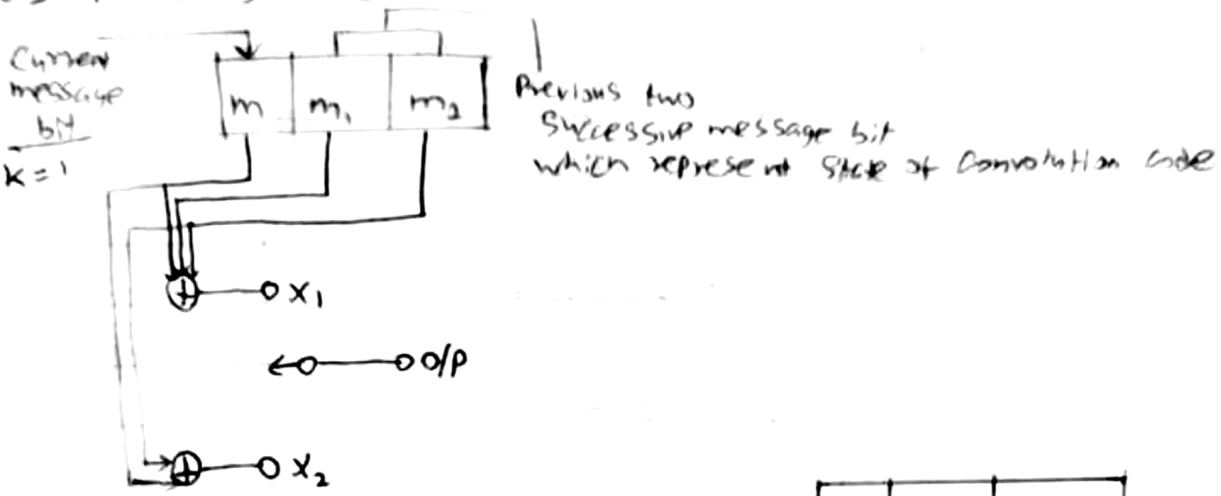
Lecture 10 Convolution, Trellis Codes

Convolutional Codes basics, parameters & designing

In convolutional codes, block of 'n' code digits generated by the encoder in time unit depends on not only block of 'k' message digits with in that time unit but also on the preceding (m-1) blocks of message digits



$x_1 = m \oplus m_1 \oplus m_2$

$x_2 = m \oplus m_2$

m_1	m_2	State
0	0	a
0	1	b
1	0	c
1	1	d

k = no of message bits = 1

n = no of encoded o/p bits = 2

K = constarain length = 3

- Here o/p will switch in between x_1 & x_2 so output will be

$X = x_1 x_2 x_1 x_2 \ldots$

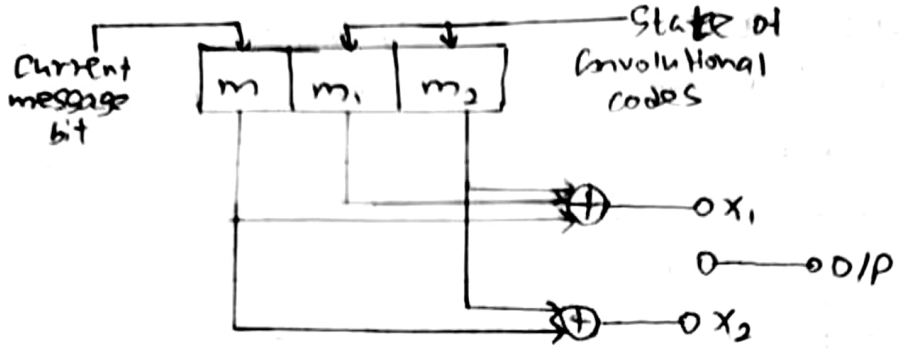
- Code rate $r = \frac{k}{n} = \frac{1}{2}$

Constraint Length (K)

- Single message bit influences encoder o/p for different successive shift

Code dimensions $(n,k) = (2,1)$

Convolution codes states & Code tree



$$X_1 = m \oplus m_1 \oplus m_2$$

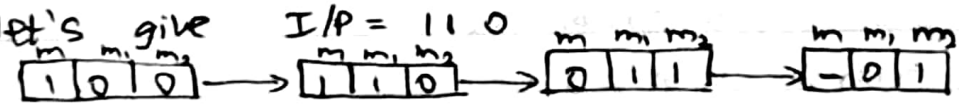
$$X_2 = m \oplus m_2$$

m_1	m_2	State
0	0	a
0	1	b
1	0	c
1	1	d

Code tree

- Each branch of tree represent an I/P symbol with the corresponding pair of O/P binary symbols indicating on the branch.

- let's give I/P = 1 1 0



$$X_1 = m \oplus m_1 \oplus m_2$$

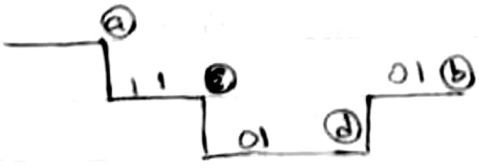
$$X_2 = m \oplus m_2$$

$X_1 = 1$, $X_2 = 1$ $\Rightarrow X_1X_2 = 11$

$X_1 = 0$, $X_2 = 1$ $\Rightarrow X_1X_2 = 01$

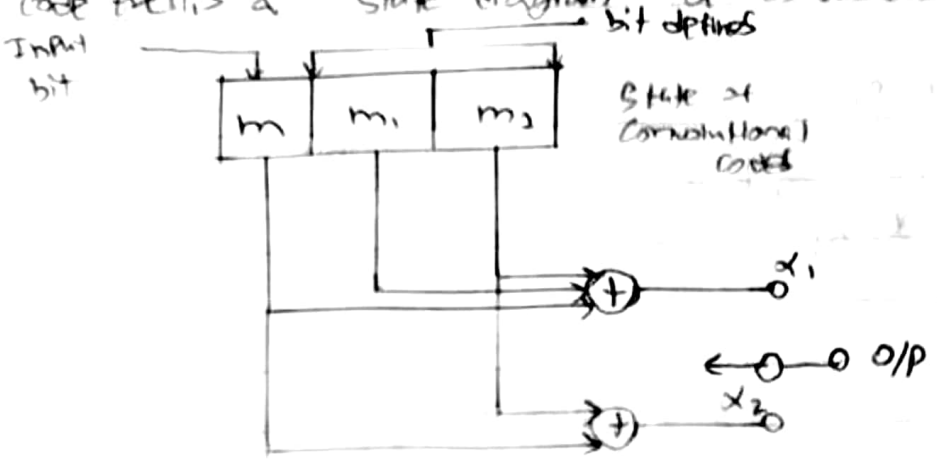
$X_1 = 0$, $X_2 = 1$ $\Rightarrow X_1X_2 = 01$

State-a State-c state-d State-b



- down step means I/P = 1
- up step means I/P = 0

Code trellis & State diagram of Convolutional codes



m_1	m_2	State
0	0	a
0	1	b
1	0	c
1	1	d

$x_1 = m \oplus m_1 \oplus m_2$

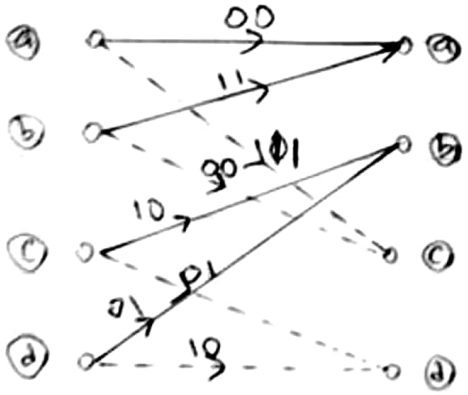
$x_2 = m \oplus m_2$

m	m_1	m_2	x_1	x_2	Current State	Next State
0	0	0	0	0	a	a
1	0	0	1	1	a	c
0	0	1	1	1	b	a
1	0	1	0	0	b	c
0	1	0	1	0	c	b
1	1	0	0	1	c	d
0	1	1	0	1	d	b
1	1	1	1	0	d	d

$x_1 = m \oplus m_1 \oplus m_2$

$x_2 = m \oplus m_2$

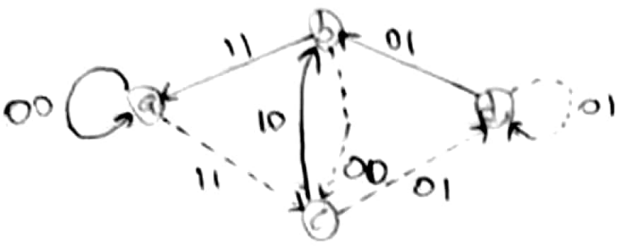
Code trellis



Input bit → 0

Input bit → 1

State diagram



→ Input 0 bit
--> Input 1 bit

Lecture 11: Viterbi Algorithm

Basics of Viterbi Algorithm

- It is a method of decoding convolution codes.
- Here we use trellis decoder received data to decode received data

Objective of Viterbi Algorithm

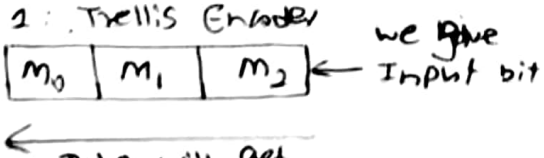
- Objective: To find the best path through the trellis that is closest to the received data bit sequence.
-

Process of Viterbi Algorithm

- Following steps are to be followed to solve Viterbi Algorithm
  - Make Trellis encoder
  - From Trellis encoder make trellis diagram
  - Match weightage with respect to trellis diagram
  - Find minimum weightage path on trellis

Example of Viterbi Algorithm

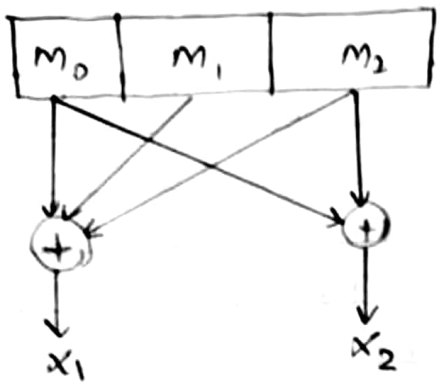
Step 1: Trellis Encoder



State	
m_0	m_1

m_0 & m_1 shows state of the Encoder

with respect to frequency of clock. The input data will get shifted in the direction from m_2 to m_0



$x_1 = m_0 \oplus m_1 \oplus m_2$

$x_2 = m_0 \oplus m_2$

Next Input bit

	m_0	m_1	m_2	x_1	x_2
a	0	0	0	0	0
	0	0	1	1	1
b	0	1	0	1	0
	0	1	1	0	1
c	1	0	0	1	1
	1	0	1	0	0
d	1	1	0	0	1
	1	1	1	1	0

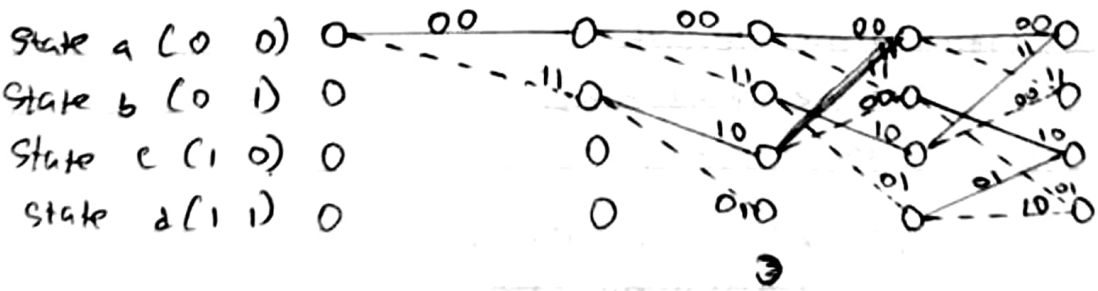
$m_0 m_1$ shows current state

$m_1 m_2$ shows next state

Step 2: Trellis Diagram

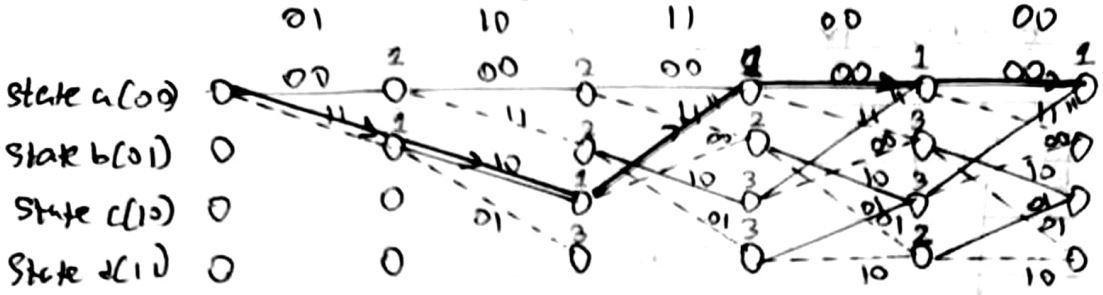
——— Input = 0

- - - - - - Input = 1



Received Data bit sequence is given by 01 10 11 00 00.

Step 3: match Weighted with respect to Trellis Diagram



* Received Data bit sequence is given by 01 10 11 00 00
* Decoded Data bit sequence is given by 11 10 11 00 00

• Here 1 bit is having an error. This is what we can corrected by having error correction code.

• viterbi algorithm, we don't correct data, we are delve to decode data.

* So, there is error in one bit during decoding by viterbi algorithm.

Turbo code

If we want to increase the length of the convolutional code, we can ~~increase using~~ do it with the help of Turbo code we increase the length to make the signal more random Then the signal is more secure.

Turbo code: more length
more randomness
More secure

→ Parallel concatenation of convolutional codes