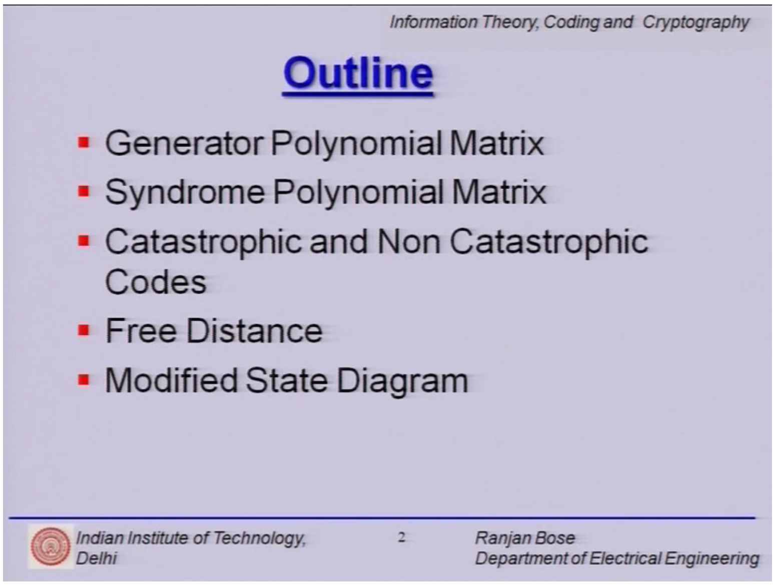
Information Theory, Coding and Cryptography

Dr. Ranjan Bose
Department of Electrical Engineering
Indian Institute of Technology, Delhi

Module - 27
Convolutional Codes
Lecture - 27

Hello and welcome to our next module on Convolutional Codes, Let us start with a brief outline.

(Refer Slide Time: 00:34)



Information Theory, Coding and Cryptography

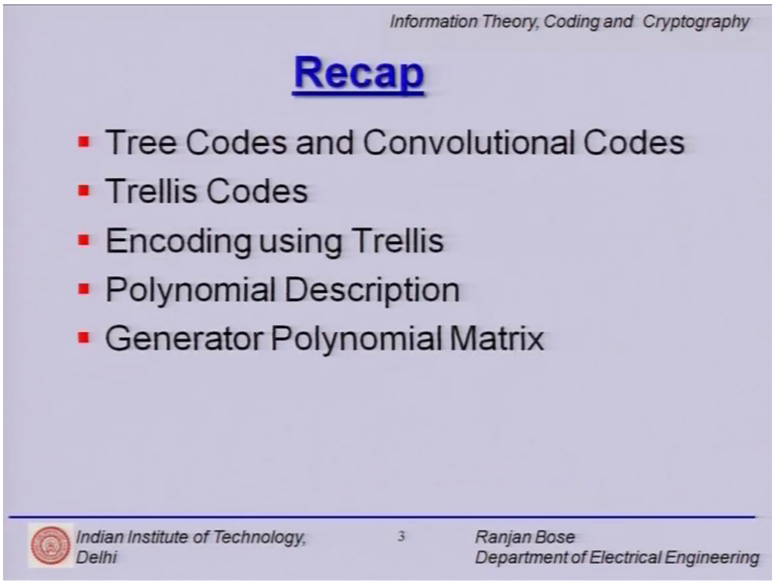
Outline

- Generator Polynomial Matrix
- Syndrome Polynomial Matrix
- Catastrophic and Non Catastrophic Codes
- Free Distance
- Modified State Diagram

Indian Institute of Technology, Delhi 2 Ranjan Bose Department of Electrical Engineering

What we will do is, we will begin with the generator polynomial matrix, for convolutional codes, followed by the syndrome polynomial matrix, then we will look at two interesting subclasses the catastrophic and non catastrophic codes for convolutional codes, then we will talk about some distance properties and figure out how good the code is. And, then we will look at a very interesting mathematical method modified state diagram which will help us analyze the distance properties very quickly.

(Refer Slide Time: 03:09)



Information Theory, Coding and Cryptography

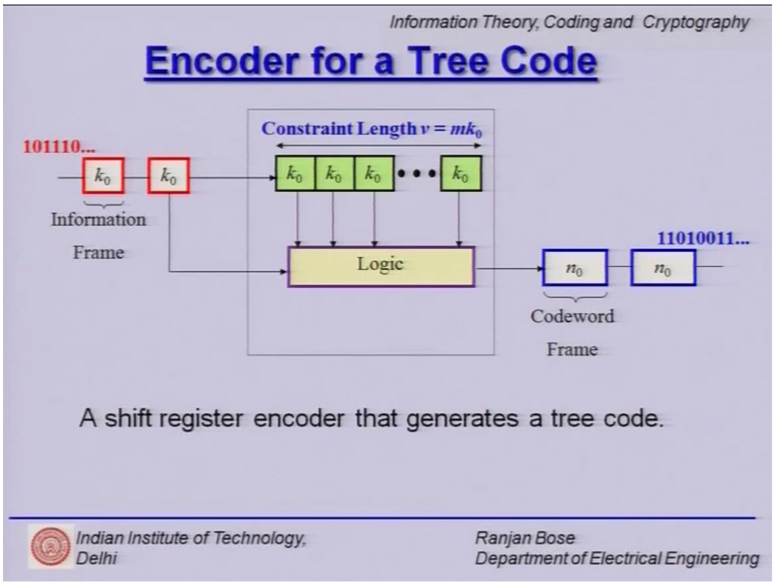
Recap

- Tree Codes and Convolutional Codes
- Trellis Codes
- Encoding using Trellis
- Polynomial Description
- Generator Polynomial Matrix

Indian Institute of Technology, Delhi 3 Ranjan Bose Department of Electrical Engineering

Let us start with a quick recap as to what we have done already, we have looked at the definition of tree codes and convolutional codes, we figured out that there is a very efficient way to represent the convolutional codes using that trellis diagram. Then we would we looked at the trellis for encoding a given bit stream and, then we would we looked at the initial polynomial description of trellis codes.

(Refer Slide Time: 01:40)



Information Theory, Coding and Cryptography

Encoder for a Tree Code

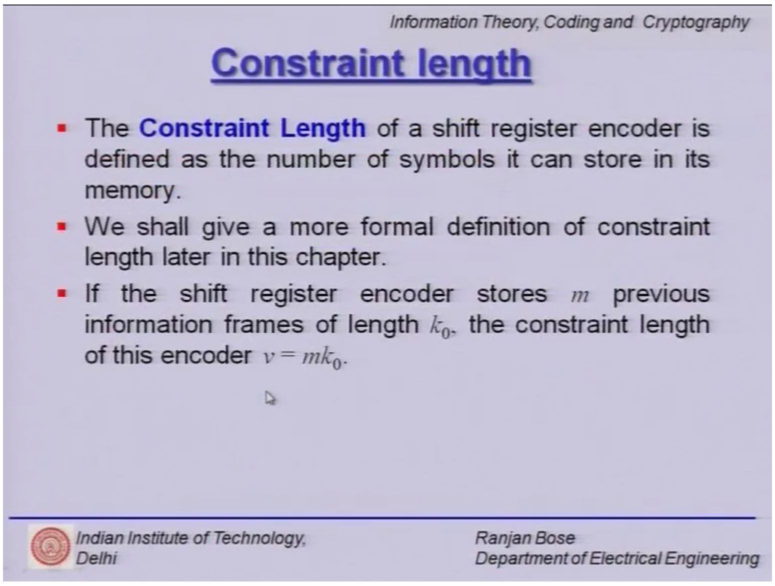
A shift register encoder that generates a tree code.

Indian Institute of Technology, Delhi Ranjan Bose Department of Electrical Engineering

Let us look at what we did last time. So, this is the general anatomy of a tree code and, when the tree code is linear then we would kind of look at it as a convolutional code, but the basic structure is as follows, there is a shift register with a certain constraint length and a logic, together they comprise the core of the encoder.

Now, we have the information frames coming in and, which are stored in this shift register. And, the based on the current storage within the shift register as well as the input information frame. We compute based on this logic the codeword frame, which is moved out. The codeword frame size is larger than the information frame size.

(Refer Slide Time: 02:38)



Information Theory, Coding and Cryptography

Constraint length

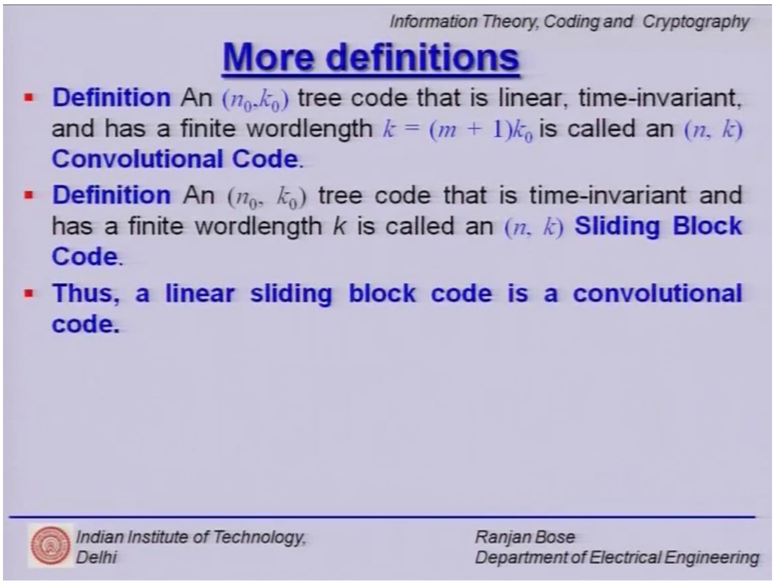
- The **Constraint Length** of a shift register encoder is defined as the number of symbols it can store in its memory.
- We shall give a more formal definition of constraint length later in this chapter.
- If the shift register encoder stores m previous information frames of length k_0, the constraint length of this encoder $v = mk_0$.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

We looked at the constraint length of the shift register and coder and, we defined as the number of symbols it can stored in it is memory.

(Refer Slide Time: 02:46)



Information Theory, Coding and Cryptography

More definitions

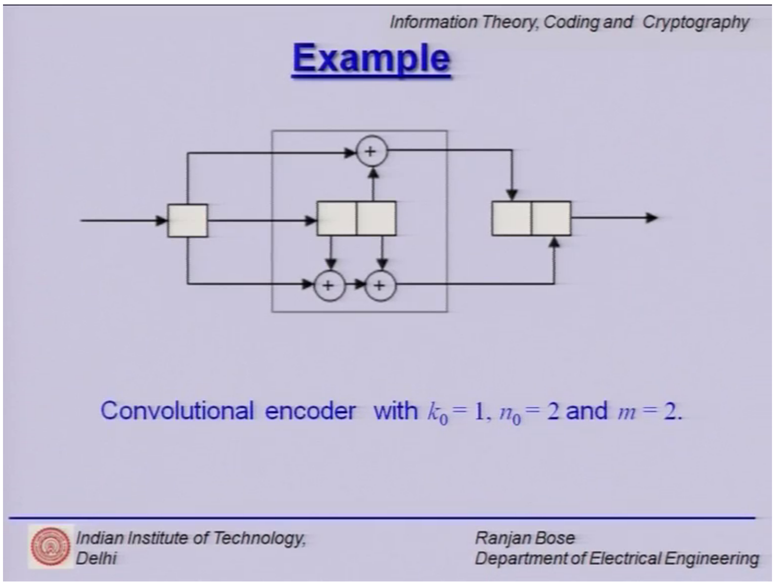
- **Definition** An (n_0,k_0) tree code that is linear, time-invariant, and has a finite wordlength $k = (m + 1)k_0$ is called an (n, k) **Convolutional Code**.
- **Definition** An (n_0, k_0) tree code that is time-invariant and has a finite wordlength *k* is called an (n, k) **Sliding Block Code**.
- **Thus, a linear sliding block code is a convolutional code.**

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

And we looked at these definitions of a convolutional code which is linear and time invariant free code. And, we talked about a sliding block code as a tree code that is time invariant and has a finite word length k and we looked at some examples.

(Refer Slide Time: 03:00)



Information Theory, Coding and Cryptography

Example

Convolutional encoder with $k_0 = 1$, $n_0 = 2$ and $m = 2$.

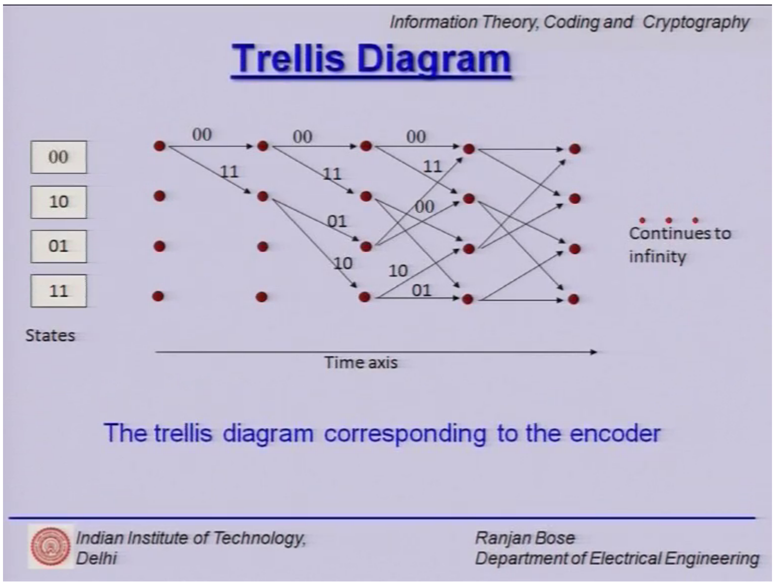
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, if you look at a very simple example of a convolutional code, we have within this dotted line 2 bit shift register, that is the memory part, the logic part is represented by this 1 2 and 3 exhorting functions binary addition without carrying.

So, 1 bit comes in and 2 bits are sent out of this encoder. So, it is a rate 1 by 2 simple convolutional encoder.

(Refer Slide Time: 03:35)



Information Theory, Coding and Cryptography

Trellis Diagram

The trellis diagram corresponding to the encoder

Indian Institute of Technology, Delhi

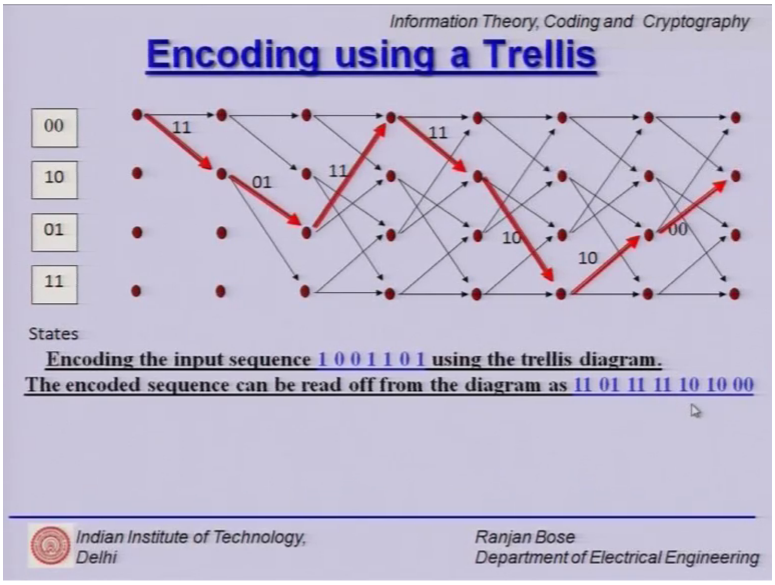
Ranjan Bose
Department of Electrical Engineering

Now the same conversion encoder can be equivalently represented using it is trellis diagram, where if you see the 2 states in the memory represents four possible states. So, the 2 inputs in the memory the 2 bit memory can either be a 0 0 0 1 1 0 and 1 1. So, these are the four possible states and that is it we cannot have any more states, if the number of memory elements is 2.

Now, based on what is coming in either there is a state transition or there is no transition. So, we keep juggling between these 4 states on the X axis is that time, where in for every input bit that comes in we either make a state transition, or not and then we wait for the next input and, we do a computation and, whatever is the output the 2 bit output is written on top of the arrows. So, this trellis diagram in its entirety represents the convolutional encoder.

So, if you give me a trellis diagram I will write the circuit for this convolution encoder and, if you give me the circuit diagram, I will be able to generate the trellis diagram.

(Refer Slide Time: 04:55)



Information Theory, Coding and Cryptography

Encoding using a Trellis

Encoding the input sequence 1 0 0 1 1 0 1 using the trellis diagram.

The encoded sequence can be read off from the diagram as 11 01 11 11 10 10 00

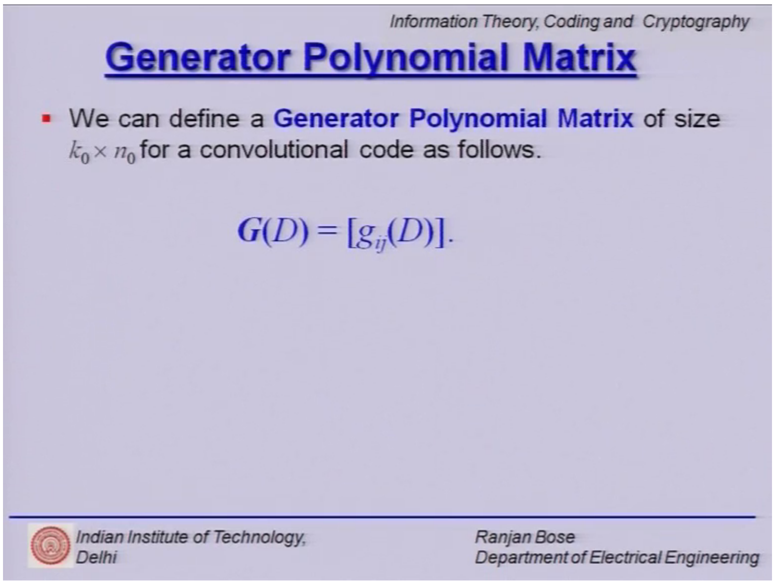
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, encoding was very easy we saw in the last class, that suppose we had to encode this input sequence 1 0 0 1 1 0 1, what we do is whenever input is 1, we take the lower branch whenever the input is 0, we take the upper branch. So, this input sequence is actually giving us directions as to how to move inside this trellis because, at every node we have to take a decision whether we go up or go down. So, 1 0 0 1 1 0 1 is nothing, but down up up down down up down. So, that is the path in the trellis.

So, there is a 1 to 1 correspondence between the input sequence and a traversed path in the trellis. Now, whatever is written on the branches is the output of the encoder at that time. So, the encoded bit stream is very simply written out as just reading out, what is written on top of the branches; so 1 1, 0 1, 1 1, 1 1, 1 0, 1 0, 0 0 that is my output right here. So, it is very easy to encode. It also brings home a very important point that decoding should tantamount to finding the most likely path in the trellis. So, this is what we have actually encoded and sent out.

(Refer Slide Time: 06:20)



Information Theory, Coding and Cryptography

Generator Polynomial Matrix

- We can define a **Generator Polynomial Matrix** of size $k_0 \times n_0$ for a convolutional code as follows.

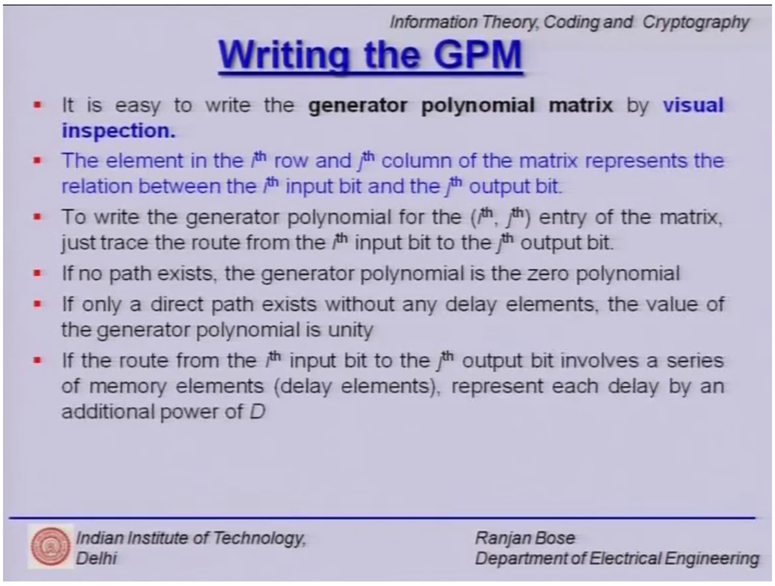
$$G(D) = [g_{ij}(D)].$$

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, we also looked at the generator polynomial matrix, which is just linking the inputs to the outputs through the delays.

(Refer Slide Time: 06:30)



Information Theory, Coding and Cryptography

Writing the GPM

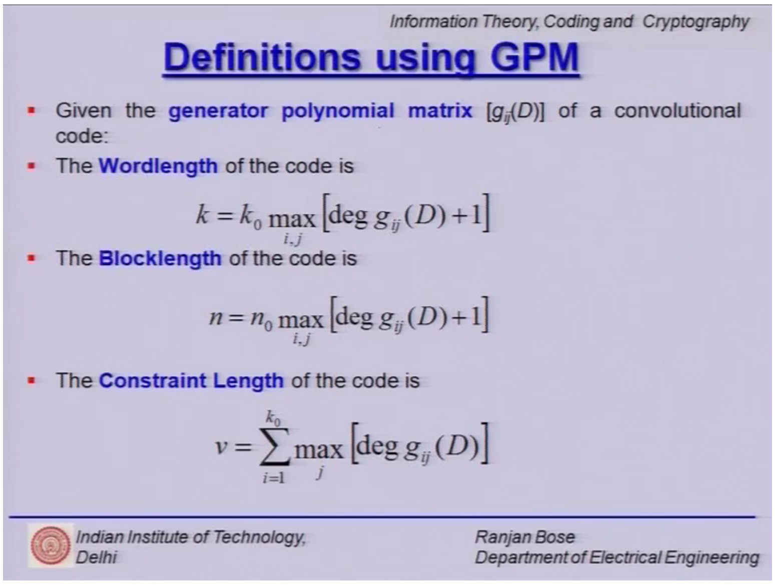
- It is easy to write the **generator polynomial matrix** by **visual inspection.**
- The element in the i^{th} row and j^{th} column of the matrix represents the relation between the i^{th} input bit and the j^{th} output bit.
- To write the generator polynomial for the (i^{th}, j^{th}) entry of the matrix, just trace the route from the i^{th} input bit to the j^{th} output bit.
- If no path exists, the generator polynomial is the zero polynomial
- If only a direct path exists without any delay elements, the value of the generator polynomial is unity
- If the route from the i^{th} input bit to the j^{th} output bit involves a series of memory elements (delay elements), represent each delay by an additional power of D

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, how do we write the generator polynomial matrix using visual inspection? What we do is the elements of i-th row and j-th column is just the relationship between the i-th input bit and the j-th output bit. So, if they are not connected then it is 0, if they are connected directly it is 1, if it is connected through delays, we count the number of delays and add accordingly.

(Refer Slide Time: 06:55)



Information Theory, Coding and Cryptography

Definitions using GPM

- Given the **generator polynomial matrix** $[g_{ij}(D)]$ of a convolutional code:
- The **Wordlength** of the code is

$$k = k_0 \max_{i,j} \left[\deg g_{ij}(D) + 1\right]$$

- The **Blocklength** of the code is

$$n = n_0 \max_{i,j} \left[\deg g_{ij}(D) + 1\right]$$

- The **Constraint Length** of the code is

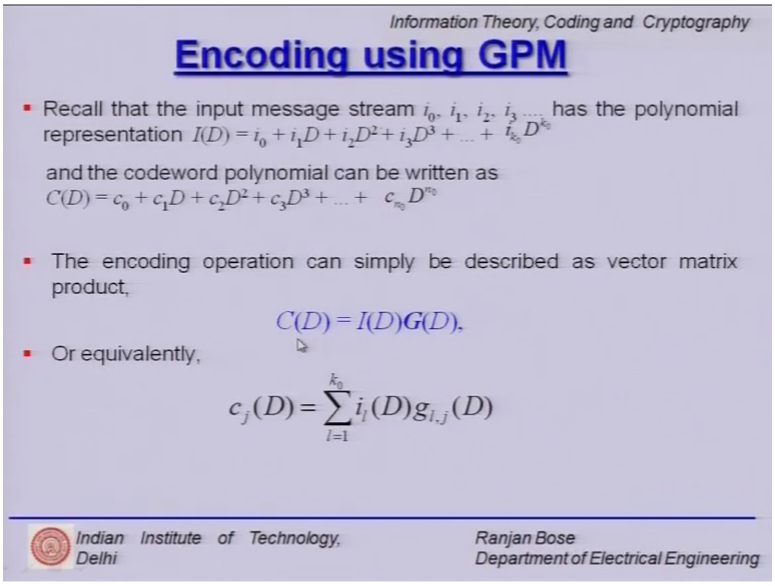
$$v = \sum_{i=1}^{k_0} \max_{j} \left[\deg g_{ij}(D)\right]$$

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, we have some definitions using the generator polynomial matrix, we talk about the word length is defined using the degree of this generator polynomial matrix as follows, similarly the block length of the code and the constraint length, all can be defined in terms of the degree of g ij and it is max or all the possible rows.

(Refer Slide Time: 07:21)



Information Theory, Coding and Cryptography

Encoding using GPM

- Recall that the input message stream $i_0, i_1, i_2, i_3 \ldots$ has the polynomial representation $I(D) = i_0 + i_1D + i_2D^2 + i_3D^3 + \ldots + i_{k_0}D^{k_0}$

  and the codeword polynomial can be written as
  $C(D) = c_0 + c_1D + c_2D^2 + c_3D^3 + \ldots + c_{n_0}D^{n_0}$

- The encoding operation can simply be described as vector matrix product,

$$C(D) = I(D)\boldsymbol{G}(D),$$

- Or equivalently,

$$c_j(D) = \sum_{l=1}^{k_0} i_l(D) g_{l,j}(D)$$

Indian Institute of Technology, Delhi

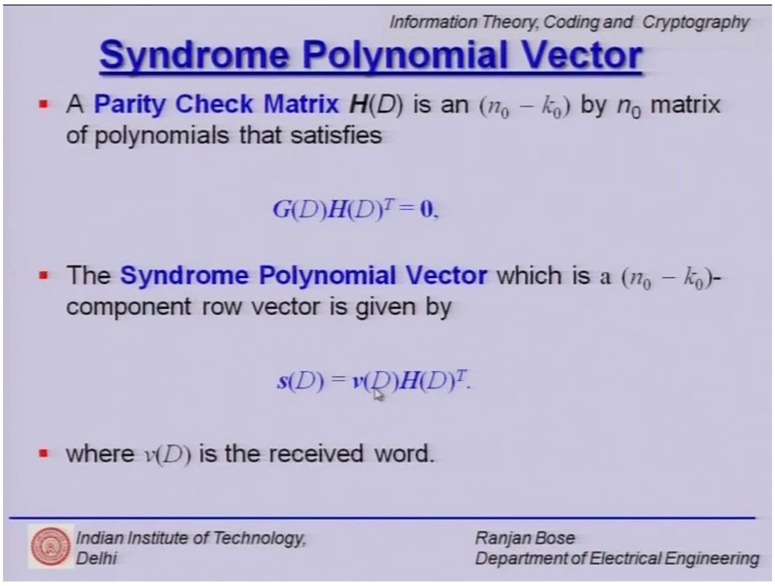
Ranjan Bose
Department of Electrical Engineering

Now, we can use this generator polynomial matrix GPM to encode, how do we do that it is pretty simple any input bit stream can be represented as i 0 i 1 i 2 i 3 and so, and so forth and equivalently in terms of the delays, so, we can use this indeterminate D. So, i 0 plus i 1 D plus i 2 D squared i 3 D cubed and so, and so forth.

Because please note for every clock cycle we go to the next input bit. So, it is in terms of the delays that we can write very efficiently. So, the encoded code word polynomial can also be written as c 0, c 1, D c to D squared and so and so forth. And what we can write is a c D is nothing, but this information bits polynomial into this G D.

So, equivalently we have any particular c j D is nothing but the summation I l D g l j D summation. So, basically we have a polynomial input multiplied with this generator polynomial matrix to give a polynomial output ok. So, we have this representation very clear in my mind. And, we would be immediately looking for an equivalent parity check polynomial and consequently the syndrome polynomial.

(Refer Slide Time: 08:50)



Information Theory, Coding and Cryptography

Syndrome Polynomial Vector

- A **Parity Check Matrix** $\boldsymbol{H}(D)$ is an $(n_0 - k_0)$ by n_0 matrix of polynomials that satisfies

$$G(D)\boldsymbol{H}(D)^T = \mathbf{0},$$

- The **Syndrome Polynomial Vector** which is a $(n_0 - k_0)$-component row vector is given by

$$\boldsymbol{s}(D) = \boldsymbol{v}(D)\boldsymbol{H}(D)^T.$$

- where $v(D)$ is the received word.

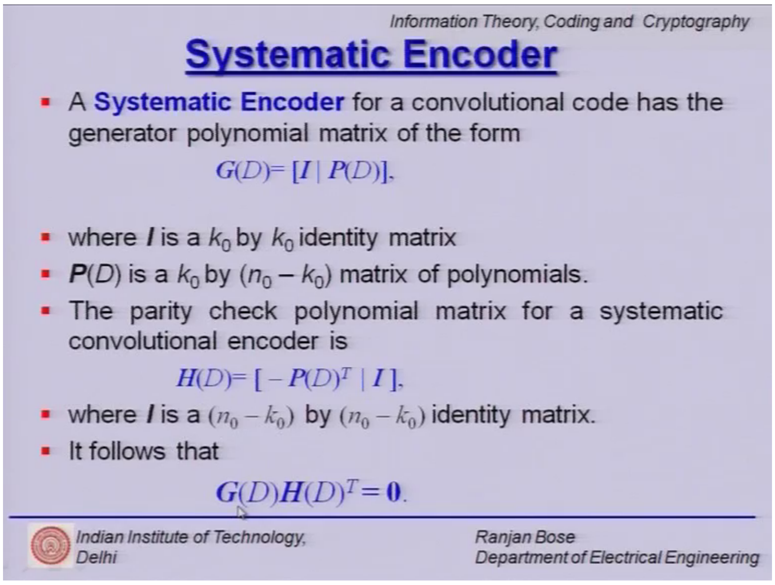
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, if you look at the parity check matrix H of D it is an n naught minus k naught by n naught matrix, which must satisfy the condition G D into H D transpose equal to 0, this theory was developed in terms of just the generator matrix and, the polynomial check matrix in linear block codes.

So, we have the syndrome polynomial vector, which is S as a function of D is nothing, but the received vector new D times H D transpose right, we have done this similar theory earlier also. So, all of that theory is equal is applicable here.

(Refer Slide Time: 09:35)



Information Theory, Coding and Cryptography

Systematic Encoder

- A **Systematic Encoder** for a convolutional code has the generator polynomial matrix of the form

$$G(D) = [I \mid P(D)],$$

- where I is a k_0 by k_0 identity matrix
- $P(D)$ is a k_0 by $(n_0 - k_0)$ matrix of polynomials.
- The parity check polynomial matrix for a systematic convolutional encoder is

$$H(D) = [-P(D)^T \mid I],$$

- where I is a $(n_0 - k_0)$ by $(n_0 - k_0)$ identity matrix.
- It follows that

$$G(D)H(D)^T = \mathbf{0}.$$

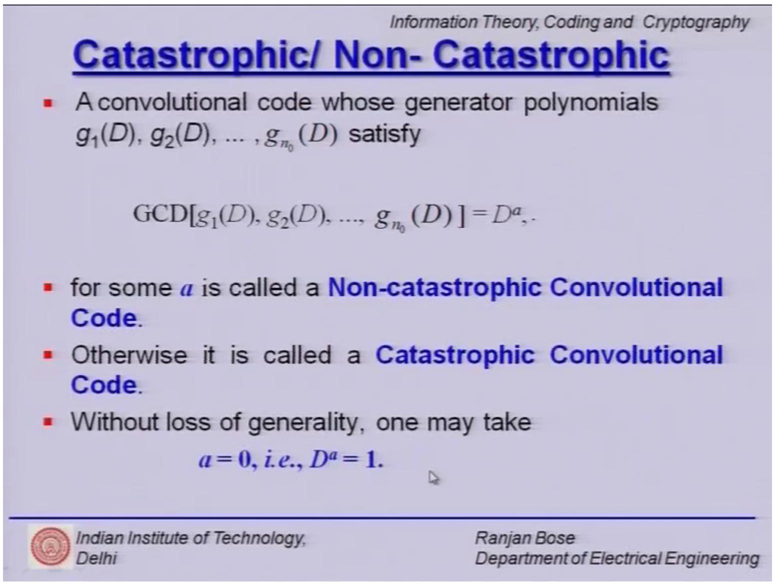
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, we have the notion of a systematic encoder for a convolutional code, which has the form G D equal to their identity matrix and the parity matrix P T.

Now, P D is the matrix of the polynomials and, the parity check polynomial matrix H of D can be quickly written in terms of minus P D transpose and I. So, you can directly see from this construction the G D into H D transpose is equal to 0 that is my construction.

(Refer Slide Time: 10:22)



Information Theory, Coding and Cryptography

Catastrophic/ Non- Catastrophic

- A convolutional code whose generator polynomials $g_1(D), g_2(D), \ldots, g_{n_0}(D)$ satisfy

$$\mathrm{GCD}[g_1(D), g_2(D), \ldots, g_{n_0}(D)] = D^a,.$$

- for some a is called a **Non-catastrophic Convolutional Code**.
- Otherwise it is called a **Catastrophic Convolutional Code**.
- Without loss of generality, one may take

$$a = 0, \text{ i.e., } D^a = 1.$$

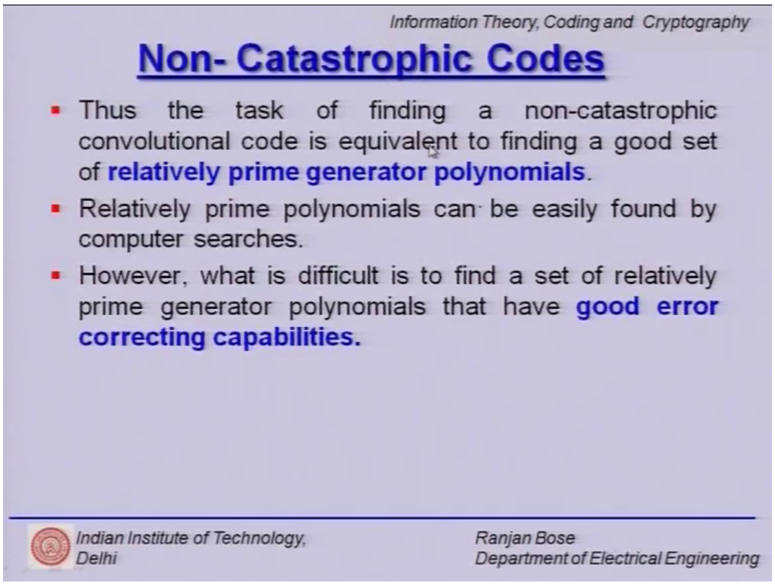
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, let us define two kinds of convolutional encoders catastrophic and non catastrophic. A convolutional code whose generator polynomials g 1 D g 2 D and so and so forth, which satisfies the GCD, the greatest common divisor g 1 D comma g 2 D so, and so forth up to g n not D equal to some D raise for alpha, or a where some a if this condition is met, then we say that it is a non catastrophic convolutional code. Otherwise we say it is a catastrophic code.

So, without loss of generality this a can be taken to be 0 leading to D raised by a equal to 1. So, we have the equivalent definition of GCD of g 1 D g 2 D up to g and not D equal to 1. So, basically we are talking about relative primes. So, this g 1 D g 2 D etcetera should be relative primes.

(Refer Slide Time: 11:26)



Information Theory, Coding and Cryptography

Non- Catastrophic Codes

- Thus the task of finding a non-catastrophic convolutional code is equivalent to finding a good set of **relatively prime generator polynomials**.
- Relatively prime polynomials can be easily found by computer searches.
- However, what is difficult is to find a set of relatively prime generator polynomials that have **good error correcting capabilities.**

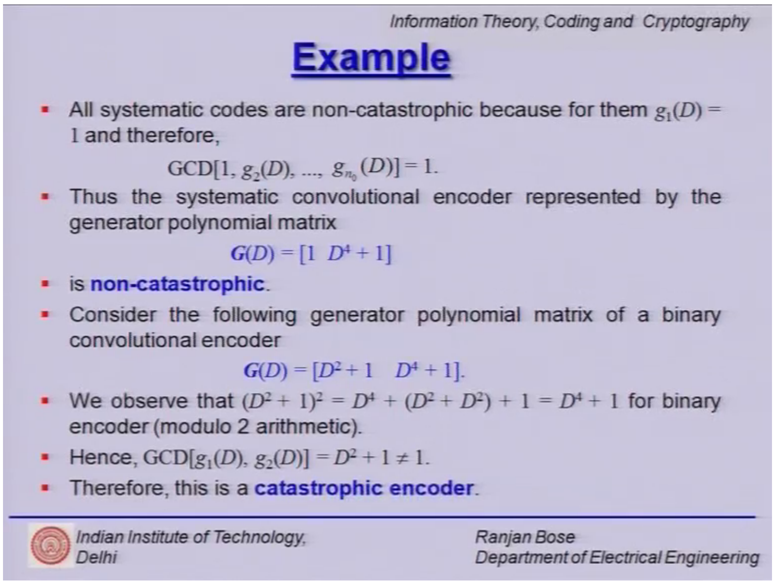
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

The does the task of finding a non catastrophic convolutional code is equivalent to finding good set of relative prime generator polynomials, that is they do not have a common factor other than 1. These relatively relative prime polynomials can be found easily by computer searches and, what is important is to have good error correcting capabilities for these non catastrophic codes, we will link these non catastrophic codes to error propagation shortly.

So, let us look at a quick example by definition all systematic codes are non catastrophic because, of the presence of the identity matrix in the beginning of the generator polynomial matrix.

(Refer Slide Time: 12:00)



Information Theory, Coding and Cryptography

Example

- All systematic codes are non-catastrophic because for them $g_1(D) = 1$ and therefore,

$$\text{GCD}[1, g_2(D), \ldots, g_{n_0}(D)] = 1.$$

- Thus the systematic convolutional encoder represented by the generator polynomial matrix

$$G(D) = [1 \quad D^4 + 1]$$

- is **non-catastrophic**.
- Consider the following generator polynomial matrix of a binary convolutional encoder

$$G(D) = [D^2 + 1 \quad D^4 + 1].$$

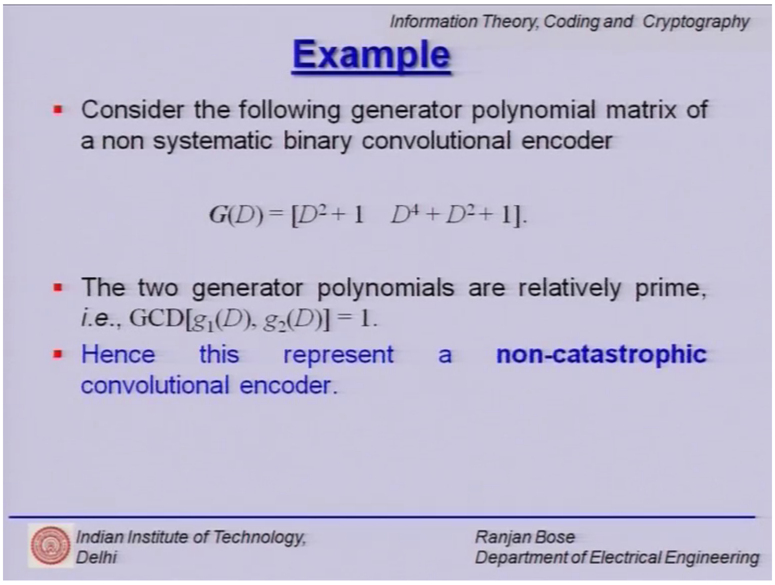
- We observe that $(D^2 + 1)^2 = D^4 + (D^2 + D^2) + 1 = D^4 + 1$ for binary encoder (modulo 2 arithmetic).
- Hence, $\text{GCD}[g_1(D), g_2(D)] = D^2 + 1 \neq 1$.
- Therefore, this is a **catastrophic encoder**.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Because, simply GCD of 1 comma g two D and so and so, forth is 1. So, if you have the simple G D equal to 1 space D raised 4 plus 1 as my generator polynomial matrix that is directly non catastrophic. But if you look at a different generator polynomial matrix, we you look at it work at it a little bit more and you find that they are not relative frames, there is a common factor and consequently it is a catastrophic conventional code leading to some kind of an error propagation.

(Refer Slide Time: 12:50)



Information Theory, Coding and Cryptography

Example

- Consider the following generator polynomial matrix of a non systematic binary convolutional encoder

$$G(D) = [D^2 + 1 \quad D^4 + D^2 + 1].$$

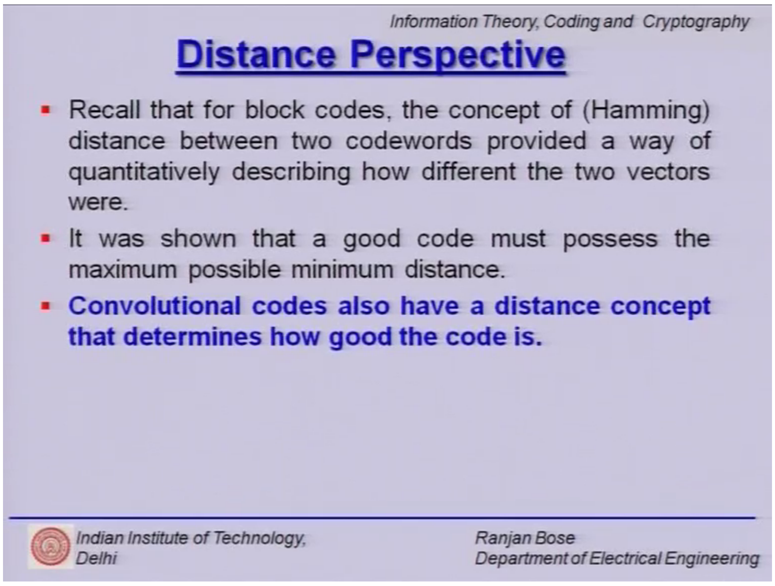
- The two generator polynomials are relatively prime, *i.e.*, $\text{GCD}[g_1(D), g_2(D)] = 1$.
- Hence this represent a **non-catastrophic** convolutional encoder.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

If you now look at another 1 G D given by this generator polynomial matrix, then you can try to again find out the GCD and, again you are left with unity leading us to believe that this is a non catastrophic convolution encoder. So, one can take many such examples and do a simple math to figure out, whether it is a catastrophic or non catastrophic convolutional encoder.

(Refer Slide Time: 13:20)



Information Theory, Coding and Cryptography

Distance Perspective

- Recall that for block codes, the concept of (Hamming) distance between two codewords provided a way of quantitatively describing how different the two vectors were.
- It was shown that a good code must possess the maximum possible minimum distance.
- **Convolutional codes also have a distance concept that determines how good the code is.**

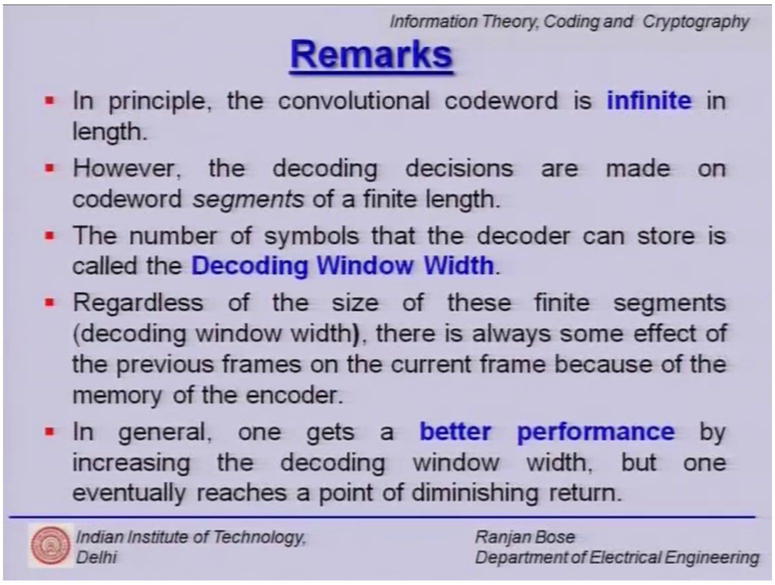
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, we move our attention to how good the code is and we know that the hamming distance properties help us characterize how good these codes are. So, do we have an equivalent definition for that, recall that the hamming distance between 2 code words provided a good way to finding out how good the error correcting capability would be because, we had linked D star to be greater than or equal to 2 t plus 1, where t was the number of errors it could correct, it was shown that good code must possess good or large minimum distance properties.

So, convolutional codes also have a distance concept, which defines how good the code is.

(Refer Slide Time: 14:13)



Information Theory, Coding and Cryptography

Remarks

- In principle, the convolutional codeword is **infinite** in length.
- However, the decoding decisions are made on codeword *segments* of a finite length.
- The number of symbols that the decoder can store is called the **Decoding Window Width**.
- Regardless of the size of these finite segments (decoding window width), there is always some effect of the previous frames on the current frame because of the memory of the encoder.
- In general, one gets a **better performance** by increasing the decoding window width, but one eventually reaches a point of diminishing return.

Indian Institute of Technology, Delhi

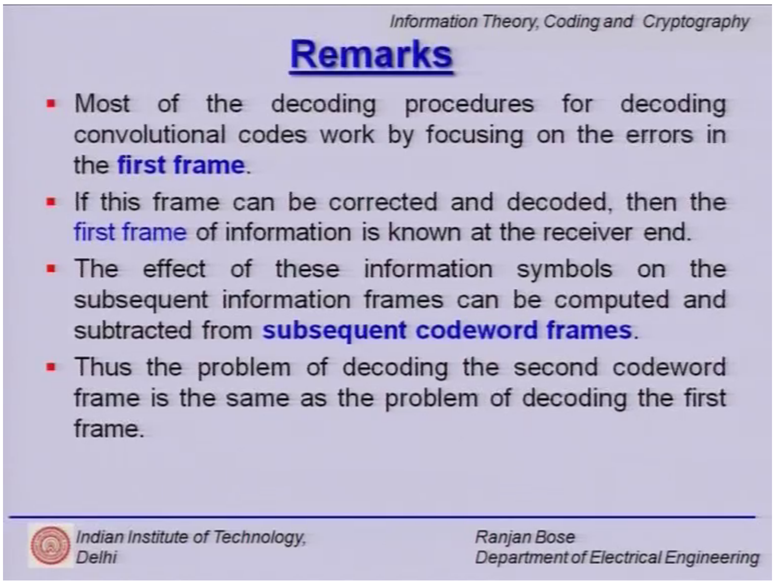
Ranjan Bose
Department of Electrical Engineering

But let us look at some basic things, there is no real notion of a codeword, why because an infinite stream of data comes in and, an infinite stream of coded data goes out. So, in technically the codeword length is in finite, but the encoding and decoding is done using frames.

And decoding is done based on a fixed segment of finite length; this is called the decoding window. So, even though we can keep going on and on we say that we have received say 40 frames and now we would like to declare the results.

So, regardless of the size of this finite segment decoding window widths, where there is always some effect of the previous frame on the current frame simply, because these are codes with memory. So, in general one gets better performance by increasing the size of the decoding window, but it comes at the cost of the decoding delay. And we also observe this law of diminishing returns that if you keep increasing it the improvement in performance marginally increases.

(Refer Slide Time: 15:33)



Information Theory, Coding and Cryptography

Remarks

- Most of the decoding procedures for decoding convolutional codes work by focusing on the errors in the **first frame**.
- If this frame can be corrected and decoded, then the first frame of information is known at the receiver end.
- The effect of these information symbols on the subsequent information frames can be computed and subtracted from **subsequent codeword frames**.
- Thus the problem of decoding the second codeword frame is the same as the problem of decoding the first frame.

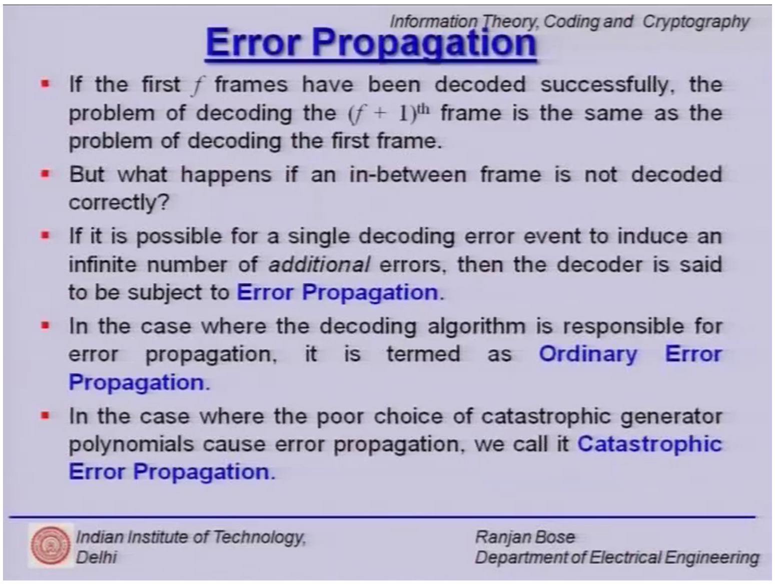
Indian Institute of Technology,
Delhi

Ranjan Bose
Department of Electrical Engineering

So, most of the decoding procedures work on focusing on the errors in the first frame, or the first few frames, if this frame can be corrected and decoded then we are in business and, the first frame of information becomes known to the receiver end.

So, what the reserve will do is use that first decoded frame please note first correctly decoded frame and the next received frame together to decode the next frame, then there it will use the first 2 frames to decode the next frame and so, on and so, forth because it is code with the memory. So, it is important to correctly decode the first frame, otherwise an error propagation might happen. So, the problem of decoding the second codeword frame is the same as the problem of decoding the first code word frame provided, you have correctly decoded the first codeword frame.

(Refer Slide Time: 16:30)



Information Theory, Coding and Cryptography

Error Propagation

- If the first f frames have been decoded successfully, the problem of decoding the $(f + 1)^{th}$ frame is the same as the problem of decoding the first frame.
- But what happens if an in-between frame is not decoded correctly?
- If it is possible for a single decoding error event to induce an infinite number of *additional* errors, then the decoder is said to be subject to Error Propagation.
- In the case where the decoding algorithm is responsible for error propagation, it is termed as Ordinary Error Propagation.
- In the case where the poor choice of catastrophic generator polynomials cause error propagation, we call it Catastrophic Error Propagation.

Indian Institute of Technology, Delhi

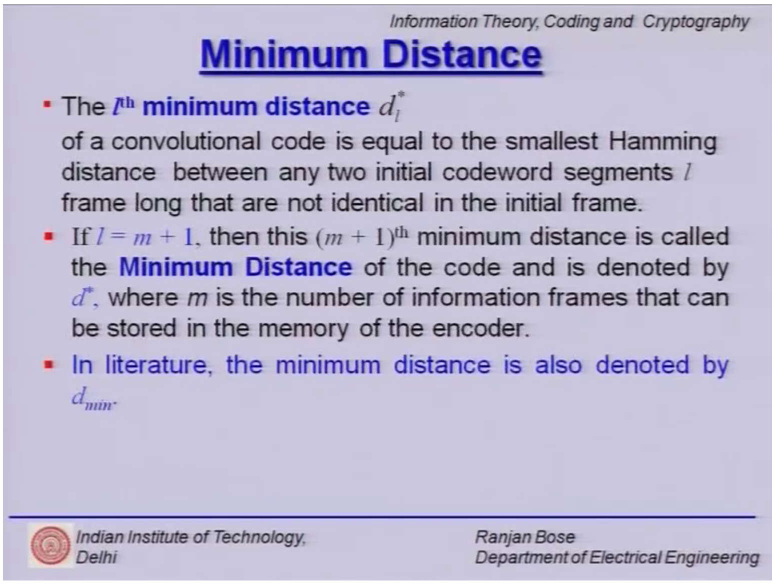
Ranjan Bose
Department of Electrical Engineering

So, let us talk quickly about error propagation, if the first f frames have been decoded successfully, the problem of decoding the f plus 1 f frame is the same as the problem of decoding the first frame, there is the basic agenda this will be used for design as well. But what happens if an happens if an in between frame is incorrectly decoded would the error propagate, will my further decisions be corrupted by what I have done incorrectly the beginning. So, we have to talk about the error propagation.

In case when the decoding algorithm is responsible for error propagation, it is called the ordinary error propagation. And, in the case when we have a poor choice of a catastrophic generator polynomials, and therefore, a catastrophic convolutional code, then we talk about it as a catastrophic error propagation.

So, ordinary error propagation and catastrophic error propagation are two distinct types of error propagation that can happen in convolutional codes.

(Refer Slide Time: 17:36)



Information Theory, Coding and Cryptography

Minimum Distance

- The l^{th} **minimum distance** d_l^* of a convolutional code is equal to the smallest Hamming distance between any two initial codeword segments l frame long that are not identical in the initial frame.
- If $l = m + 1$, then this $(m + 1)^{th}$ minimum distance is called the **Minimum Distance** of the code and is denoted by d^*, where *m* is the number of information frames that can be stored in the memory of the encoder.
- In literature, the minimum distance is also denoted by d_{min}.

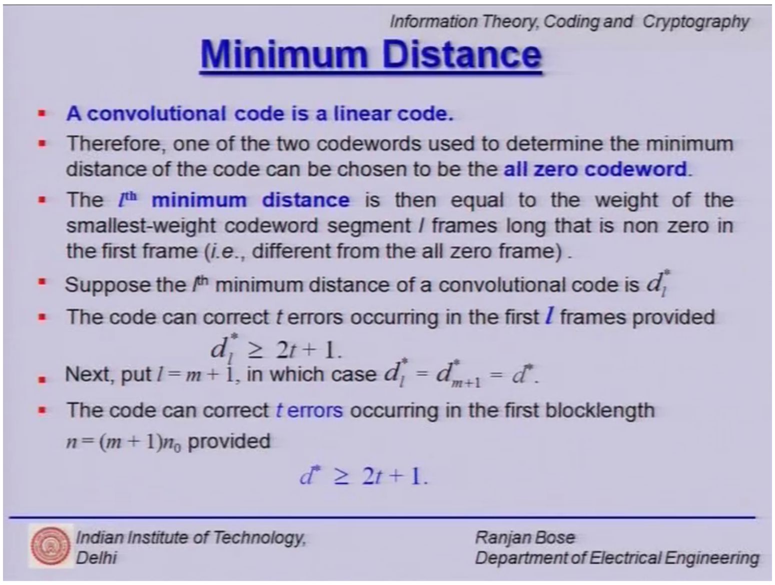
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, let us refocus on the minimum distance, or the distance properties of convolutional code. So, let us start with the l-th minimum distance denoted by D sub l and, since it is a minimum distance we put a star. So, the l-th minimum distance of a convolutional code is the smallest hamming distance, which is any 2 initial codeword segments l frame law.

So, we must specify the codeword segment lengths here, because technically speaking the codewords are of infinite length ok. So, suppose l is equal to m plus 1, then m plus one with minimum distance is called the minimum distance of the code and is denoted by D star. Where m is a number of information frame that can be stored in the memory of the encoder directly linked to the constraint length of the convolutional encoder, In literature the minimum distance is sometimes denoted by D min. So, D star and D min both find applicability in literature.

(Refer Slide Time: 18:45)



Information Theory, Coding and Cryptography

Minimum Distance

- **A convolutional code is a linear code.**
- Therefore, one of the two codewords used to determine the minimum distance of the code can be chosen to be the **all zero codeword**.
- The l^{th} **minimum distance** is then equal to the weight of the smallest-weight codeword segment l frames long that is non zero in the first frame (*i.e.*, different from the all zero frame) .
- Suppose the l^{th} minimum distance of a convolutional code is d_l^*
- The code can correct t errors occurring in the first l frames provided

$$d_l^* \ge 2t + 1.$$

- Next, put $l = m + 1$, in which case $d_l^* = d_{m+1}^* = d^*$.
- The code can correct t errors occurring in the first blocklength $n = (m + 1)n_0$ provided

$$d^* \ge 2t + 1.$$

Indian Institute of Technology, Delhi

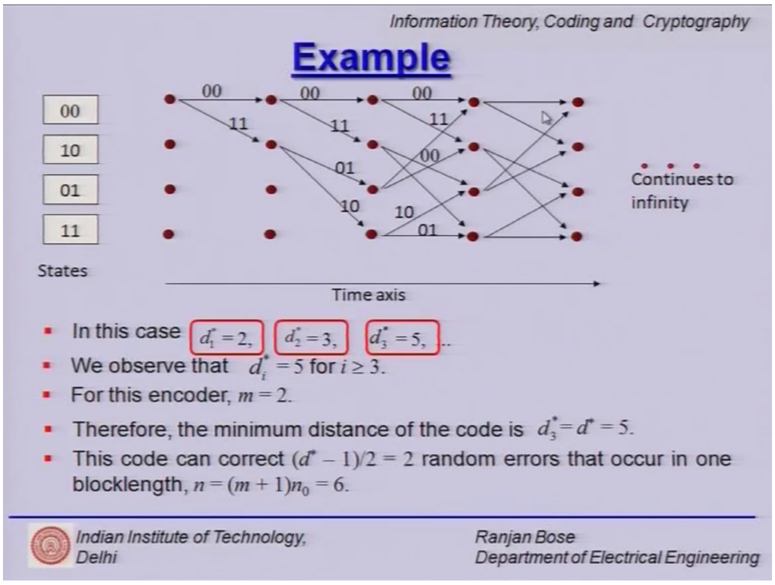
Ranjan Bose
Department of Electrical Engineering

Now, we start with this basic observation that convolutional codes are linear. So, all zero code word must be a valid code word sum of any two code words. Now, code word means segments of the code words sum of any two code words must be another code word.

So, the l-th minimum distance is equal to the weight of the smallest weight code word segment l frames long, because minimum distance translates to minimum weight. And if you do have that then d l star. So, for those l frame long segment we have greater than or equal to 2 t plus 1 as a constraint to correct t errors in those l frames.

So, code word segment of l frames is what we are talking about. So, t again denotes the number of errors it can correct.

(Refer Slide Time: 19:44)



Information Theory, Coding and Cryptography

Example

- In this case $\boxed{d_1^* = 2,}$ $\boxed{d_2^* = 3,}$ $\boxed{d_3^* = 5,}$ ...
- We observe that $d_i^* = 5$ for $i \geq 3$.
- For this encoder, $m = 2$.
- Therefore, the minimum distance of the code is $d_3^* = d^* = 5$.
- This code can correct $(d^* - 1)/2 = 2$ random errors that occur in one blocklength, $n = (m + 1)n_0 = 6$.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Let us look at a very quick example we have seen this trellis diagram before. So, we revisit it there are 4 states in this trellis as usual the x axis is the time axis and, we start with the 0 0 state. Now, r job is to find out the minimum distance of this trellis, well it is the same as finding the minimum distance of the convolutional code of which this trellis diagram is written.

So, let us start somewhere let us start with the first time instants. So, we are now finding out the minimum distance between 2 parts well the 2 parts are just 2 branches 0 0 and 1 1. So, D 1 star is 2 because the hamming distance between these 2 is 2, but then we continue and we say ok. If you look at D 2 so, my segment is now 2 frames long fine. So, the D 2 star is now the minimum distance. Now, what is the minimum distance, we already have a burden of 2 as the hamming distance plus. So, we add next.

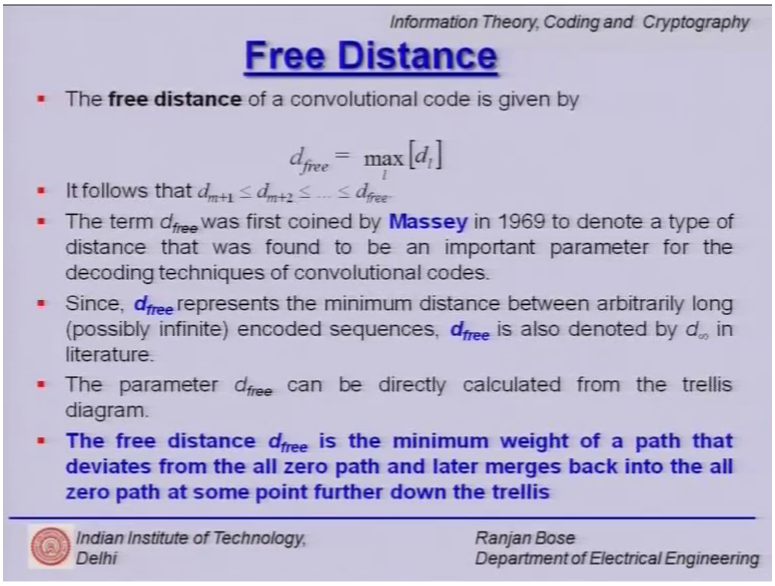
So, what if we take this all 0 path versus 0 0 and 1 1 so, in the distance between this is really not starting here. So, this and this does not constitute candidates, we have to have a path which starts here diverges and we calculate the distance. So, this first 1 0 0 and 0 0 is a candidate all 0 path. Now, we diverged here we already have a burden of 2 and, then either this or this. So, let us calculate the additional distance hamming distance obtained using the second branch.

So, comparing with the 0 0, if you see and you go to 0 1 the distance is 1. So, 2 for the first case plus one more adds up to 3, but we are not done yet there is another possibility again this is 1 0 and 0 0, again the distance is 1 on the second branch. So, 2 in the first case and 1 is in the second add up to 3. So, consequently for the first 2 frames we have D star 2 equal to 3, but we are not done yet we continue further and, we look at this third frame. Now, we are already diverged now we have 1 branch, 2 branch, 3 branch, 4.

So, we have got an exponentially growing number of paths to work with as we proceed along the trellis to two more, because we did not consider here and, now we have 4 and then this will keep growing. Again so, far the total distance up to 2 frames was 2. Now, we add here two more and then again we keep adding and we compare and we can look at and we find the D star is equal to 5.

So, we observe that this minimum distance is 5 and, if you keep going beyond that it does not increase any further because, once you have merged back to the 0 path you keep going 0 0 and so, you do not really add further. So, if you see the minimum distance has been calculated using this branch, then this branch and then going back and then it is merged back. So, no longer whether you continue to infinity, you will not keep adding anymore hamming weight consequently, you will end up with D star is equal to 5.

(Refer Slide Time: 24:00)



Information Theory, Coding and Cryptography

Free Distance

- The **free distance** of a convolutional code is given by

$$d_{free} = \max_{l} [d_l]$$

- It follows that $d_{m+1} \leq d_{m+2} \leq \ldots \leq d_{free}$
- The term d_{free} was first coined by **Massey** in 1969 to denote a type of distance that was found to be an important parameter for the decoding techniques of convolutional codes.
- Since, d_{free} represents the minimum distance between arbitrarily long (possibly infinite) encoded sequences, d_{free} is also denoted by d_∞ in literature.
- The parameter d_{free} can be directly calculated from the trellis diagram.
- **The free distance d_{free} is the minimum weight of a path that deviates from the all zero path and later merges back into the all zero path at some point further down the trellis**

Indian Institute of Technology, Delhi

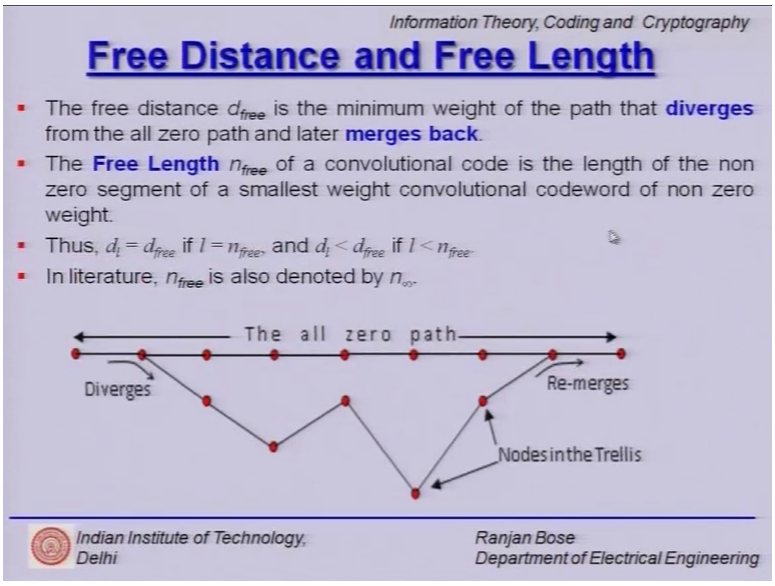
Ranjan Bose
Department of Electrical Engineering

So, the free distance of the convolutional code can be given as max over all l d l, where l was for the l segment or l frames. So, please note we have coined, or restated the words used by Massey called the free distance. Because you free to go up to infinity and calculate your distance and, whatever it is technically speaking your talking about an infinitely long codeword, you can have the d free notation.

So, d free represents the minimum distance between arbitrarily long, possibly infinite encoded sequence. So, sometimes you also denoted by d infinity, but note the d free can be directly calculated from the trellis diagram. So, the trellis diagram not only is a very efficient way to encode, the bit streams coming in but it is also important to find out the distance property of your convolutional encoder.

So, free distance d free is a minimum weight of a path that deviates from the all 0 path and later merges back. So, the key word is deviates and merges back, into the all 0 path at some point in time in the trellis, why are we talking about weight linear code. So, many hamming distance is the hamming weight.

(Refer Slide Time: 25:26)



Information Theory, Coding and Cryptography

Free Distance and Free Length

- The free distance d_{free} is the minimum weight of the path that **diverges** from the all zero path and later **merges back**.
- The **Free Length** n_{free} of a convolutional code is the length of the non zero segment of a smallest weight convolutional codeword of non zero weight.
- Thus, $d_l = d_{free}$ if $l = n_{free}$, and $d_l < d_{free}$ if $l < n_{free}$.
- In literature, n_{free} is also denoted by n_∞.

Indian Institute of Technology, Delhi

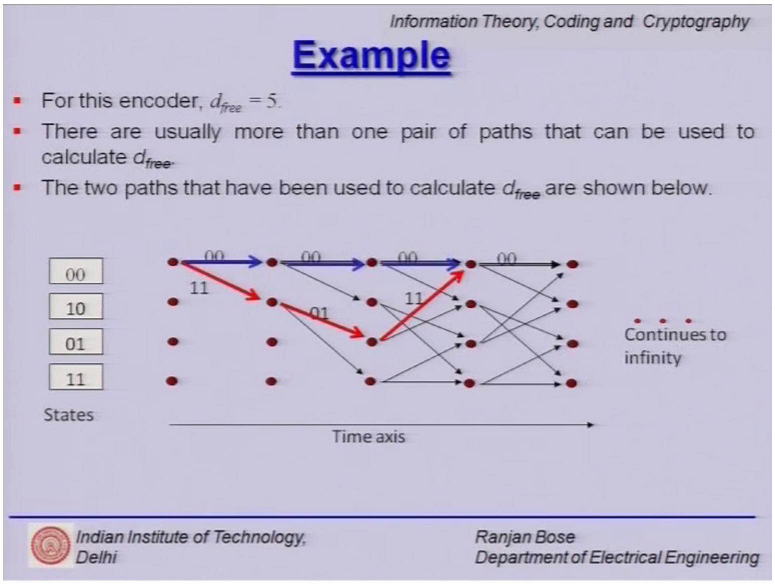
Ranjan Bose
Department of Electrical Engineering

So, what do we mean by diverges and merges back? Well, the all 0 path is shown here, that is a reference and whatever we develop here is equal for any 2 pair of paths because it is a linear code. So, you diverge at some point you keep jumping up and down in the trellis and, you merge back and then you continue merged, because then it will not give any more d increase no more hamming distance increase will happen since it has merged well.

Even though there are two paths going on, they are not contributing further to the distance.

So, you calculate branch by branch the distance keep adding and, then you have the d free. Now, we have the free length definition of a convolutional code is the length of the nonzero segment of the smallest weight convolutional code of a nonzero weight. So, it is just trying to find out how many hops there are and therefore, you have a notion of a n free coupled with t, free it is possible that n free for a larger value gives you the d free.

(Refer Slide Time: 26:43)



Information Theory, Coding and Cryptography

Example

- For this encoder, $d_{free} = 5$.
- There are usually more than one pair of paths that can be used to calculate d_{free}.
- The two paths that have been used to calculate d_{free} are shown below.

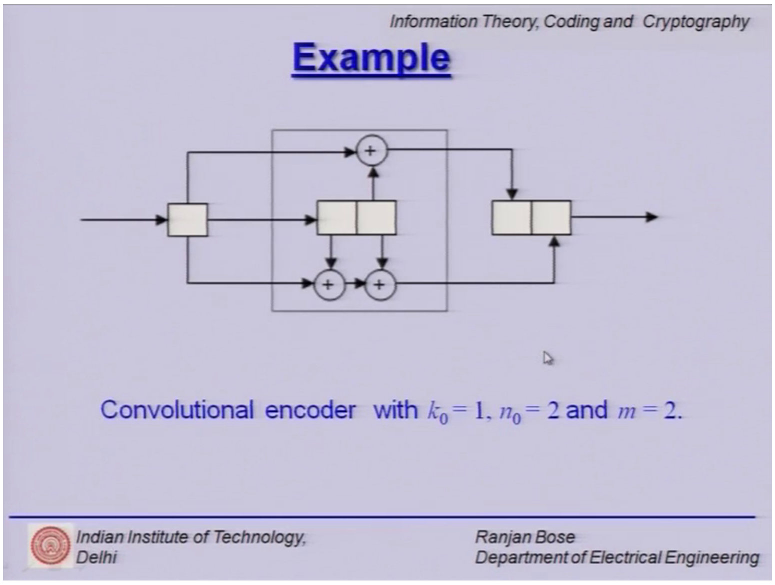
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, let us look at this example, well this is your convolutional code, I have equivalently represented it using the trellis diagram. So, here if you see we first found out the distance between 0 0 and 1 1 and, then that path diverged further and, then I have this branches between which we find out the hamming distance and, then it merges back again. So, diverges continues in the trellis merges back and, then it is free to continue together through infinity.

So, for this encoder d free is 5 you can calculate 2 plus 1 plus 2 that is 5. So, the d free for this encoder is 5. Now, d free is greater than equal to 2 t plus 1. So, clearly it can correct 1 error, please note it is a 1 by 2 encoder. So, there could be more than one pair of paths that can be used to calculate d free you can choose any one ok. So, in this example d min is equal to d free and n free is equal to 6

(Refer Slide Time: 28:07)



Information Theory, Coding and Cryptography

Example

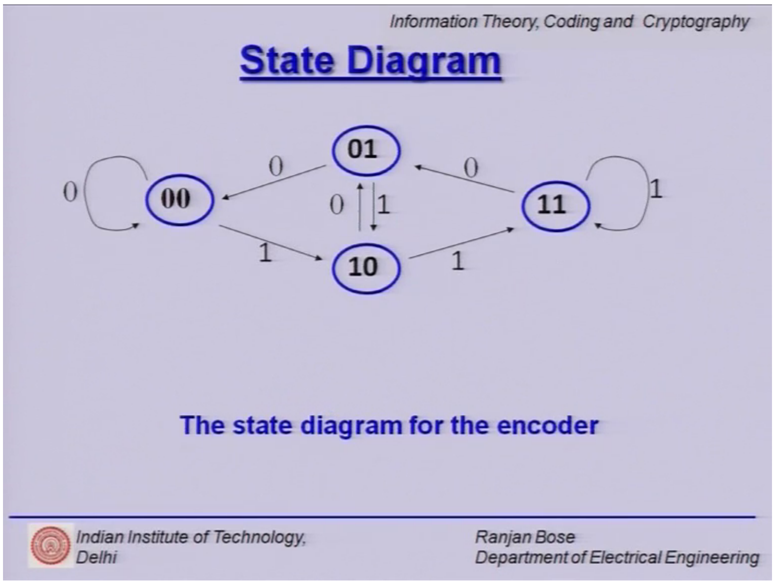
Convolutional encoder with $k_0 = 1$, $n_0 = 2$ and $m = 2$.

Indian Institute of Technology,
Delhi

Ranjan Bose
Department of Electrical Engineering

So, if you look at the encoder that we just looked at this is the simple encoder k naught equal to 1, m naught equal to 2 rate 1 by 2, encoder constraint length m is equal to 2, very simple encoder very hardware friendly, but at the same time it can correct one error. So, that is a strength of these convolutional codes, very powerful codes. The price we pay is in terms of a poorer code rate is 1 by 2 and of course, the decoding complexity which will soon encounter.

(Refer Slide Time: 28:47)



Information Theory, Coding and Cryptography

State Diagram

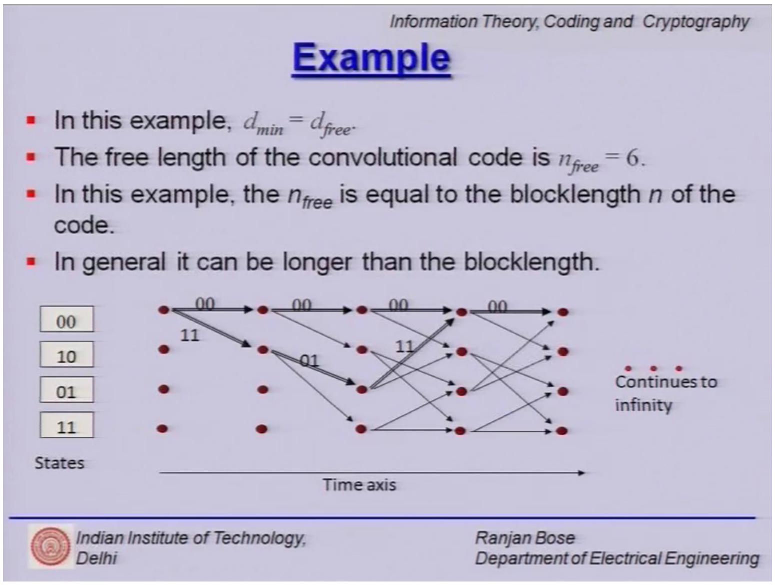
The state diagram for the encoder

Indian Institute of Technology,
Delhi

Ranjan Bose
Department of Electrical Engineering

Now, what you see is that in between the hardware representation and the trellis diagram, we first drew a state diagram, there four states and their state transitions. This is not very friendly to use, if you are going to use it to show encoding process therefore, we invented the trellis diagram. But this state diagram can be taken to a modified state diagram to give us some distance property calculations.

(Refer Slide Time: 29:19)



Information Theory, Coding and Cryptography

Example

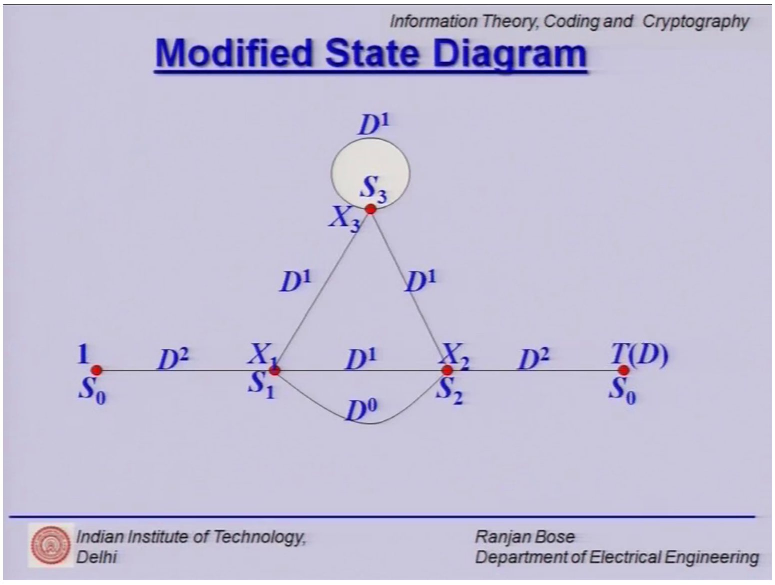
- In this example, $d_{min} = d_{free}$.
- The free length of the convolutional code is $n_{free} = 6$.
- In this example, the n_{free} is equal to the blocklength n of the code.
- In general it can be longer than the blocklength.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Why, because I am always skeptical whether I missed out any pair of paths which could have given me a smaller d free, either I write a computer program, or I can have a more efficient way to do things.

(Refer Slide Time: 29:37)



Information Theory, Coding and Cryptography

Modified State Diagram

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, we introduced this notion of a modified state diagram. So, note let us focus on this small red dots here. So, they are 1 2 3 4 5 dots they are labeled as 0 for state 0, S 1 for state 1, S 2 for state 2, S 3 for state 3. So, there are 4 states and we rewrite the S 0 state back again why do we do that, because we want to go from state S 0. If you see in the diagram here, you have these are the 4 states S 0, S 1, S 2 and S 3. You start with state S 0 and you diverge and eventually you must merge back to S 0, in between you can go through several states ok. So, this sentiment is captured in your modified state diagram, what does it show, you can start with a state S 0, you can diverge.

So, you can go to state S 1, or you can possibly go to state S 3 and, then S 2 and then and you can do as many jumps as you want, but eventually you should merge back to S 0. So, here you start with S 0 you can go to S 1, because clearly in the state diagram a transition from S 0 to S 3 is not permissible so, there is no line connecting S 0 and S 3.

Now, once you are at S 1 you can either go to S 2 or S 3. So, these are directed graphs. So, you have to have an arrow here, so, I should have an arrow here and an arrow here. Now at S 3 either you can remain in your own state S 3, or you can go from S 3 to S 2, from S 2 either I can go to S 0 I can go back to S 1. So, this modified state diagram captures all the possible transitions in that trellis diagram.

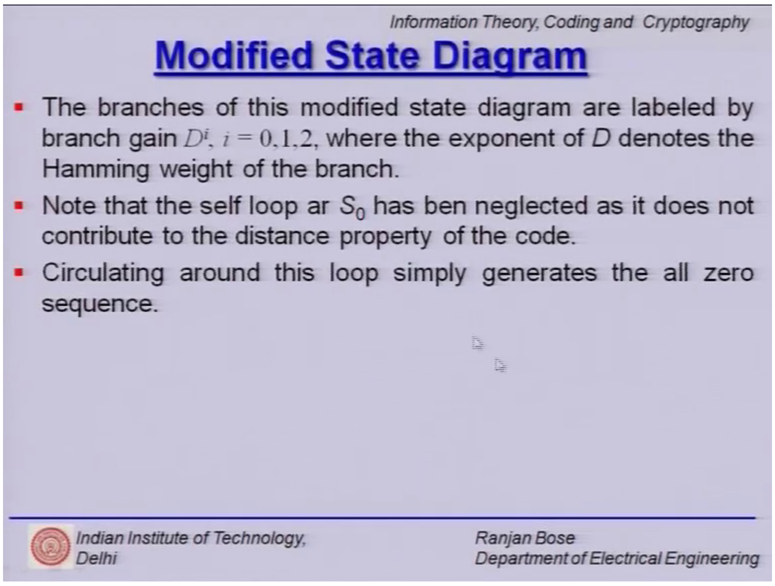
Now, what are these weights written here D 1, D 2 so, when you go from S 0 to S 1 you gather a hamming distance of 2, it is written as an exponent of D, D is that indeterminate.

So, this two represents the exponent, where is it coming from well, if you go from state S 0 to S 1, S 0 to S 1 the hamming distance that it picks up S 2.

Now, if you go from S 1 to S 2 the hamming distance it picks up is 1. So, the branch that shows a transition from S 1 to S 2 that should be labeled D raised to power 1. So, if you go from S 1 to S 2 it is labeled with D raised S power 1, why is it in the exponent because, when we look at the gain from S 0 to this S 0 which is T D, it will multiply the gains will multiply. So, the exponents will add so, your hamming distance keeps adding. So, suppose I have represent I go from S 0 to S 1 to S 2 to S 0.

So, I pick up weight 2, I pick up weight 1, I pick up weight 2 and so, the total weight picked up will be 5. So, then infinite ways I can travel from S 0 back to this S 0. So, start from diverged from 0 state merge back to 0 state, going through as many transitions as you require and then finally, you calculate the weight you pick up.

(Refer Slide Time: 33:38)



Information Theory, Coding and Cryptography

Modified State Diagram

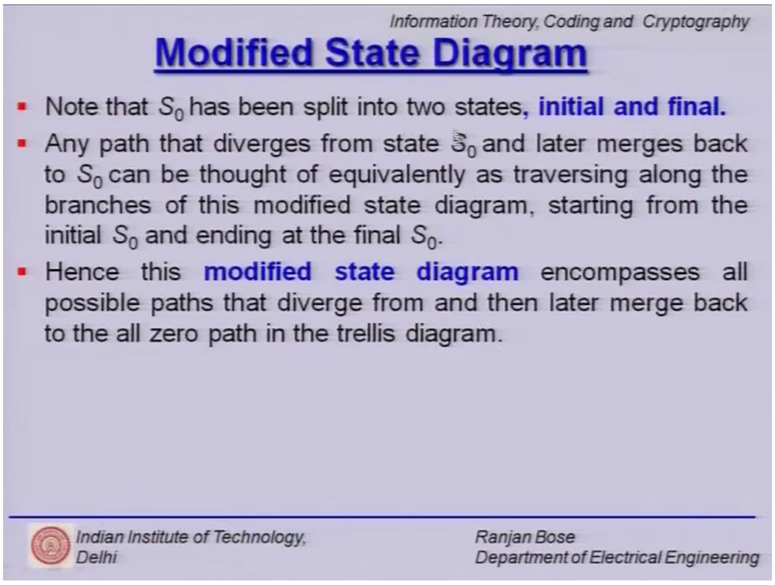
- The branches of this modified state diagram are labeled by branch gain D^i, $i = 0,1,2$, where the exponent of D denotes the Hamming weight of the branch.
- Note that the self loop ar S_0 has ben neglected as it does not contribute to the distance property of the code.
- Circulating around this loop simply generates the all zero sequence.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, the branches of the modified state diagram are labeled by the branch gain D i as I mentioned and, the self loop gain a 0 has been neglected does not contribute to the distance property.

(Refer Slide Time: 33:50)



Information Theory, Coding and Cryptography

Modified State Diagram

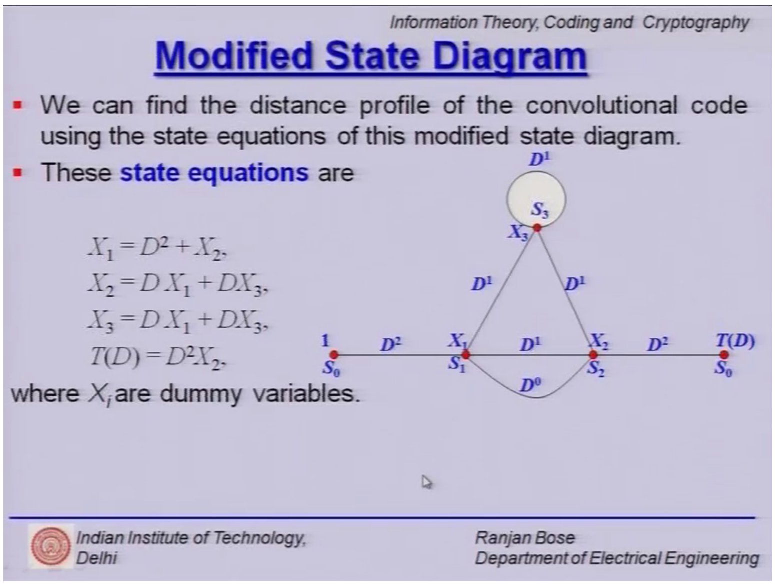
- Note that S_0 has been split into two states, **initial and final.**
- Any path that diverges from state S_0 and later merges back to S_0 can be thought of equivalently as traversing along the branches of this modified state diagram, starting from the initial S_0 and ending at the final S_0.
- Hence this **modified state diagram** encompasses all possible paths that diverge from and then later merge back to the all zero path in the trellis diagram.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, a 0 has been split into 2 states the initial and the final we go from the initial to the final picking up weight. And this is the modified state diagram that encompasses all possible paths that diverge from and, then later merge back to the all 0 path.

(Refer Slide Time: 34:07)



Information Theory, Coding and Cryptography

Modified State Diagram

- We can find the distance profile of the convolutional code using the state equations of this modified state diagram.
- These **state equations** are

$$X_1 = D^2 + X_2,$$
$$X_2 = D X_1 + D X_3,$$
$$X_3 = D X_1 + D X_3,$$
$$T(D) = D^2 X_2,$$

where X_i are dummy variables.

Indian Institute of Technology, Delhi

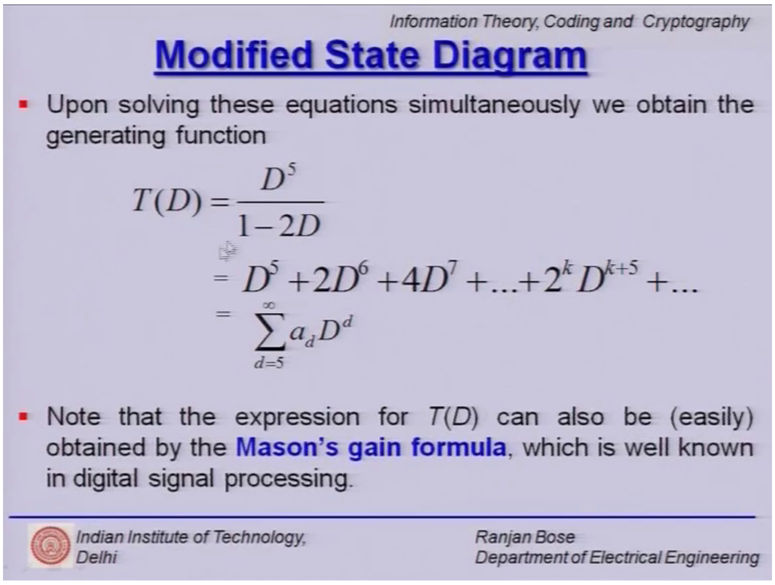
Ranjan Bose
Department of Electrical Engineering

So, how do we use it, well we write the state equations. Let us take three variables X 1, X 2 and X 3 at these 3 intermediate points, my aim is to go from one unity and reach up to T D starting from state S 0 coming back to state S 0 and, see how much weight I pick up. So, I can write X 1 write as D squared so, 1 times D squared plus there is an arrow here going back so, X 2 times weight is unity. So, D squared plus X 2 that is the state equation for X 1.

Now, X 2 has a contribution from X 3 and X 1 so, X 2 is written as D times X 1. So, X 2 is written D this is the D D times times X 1 plus D times X 3, X 3 is right here and this X 3 it is right here is coming getting a contribution X 1 times D. So, this is X 1 multiplied by D and it is a self contribution, X 3 goes loops back here. So, it is X 3 times T and finally, this D T D is X 2 times D squared that is the fourth equation.

So, we have these three intermediate variables X 1, X 2 and X 3 we have three equations we eliminate that and we get an answer of T D right, in terms of the indeterminate D.

(Refer Slide Time: 36:07)



Information Theory, Coding and Cryptography

Modified State Diagram

- Upon solving these equations simultaneously we obtain the generating function

$$T(D) = \frac{D^5}{1-2D}$$
$$= D^5 + 2D^6 + 4D^7 + \ldots + 2^k D^{k+5} + \ldots$$
$$= \sum_{d=5}^{\infty} a_d D^d$$

- Note that the expression for $T(D)$ can also be (easily) obtained by the **Mason's gain formula**, which is well known in digital signal processing.

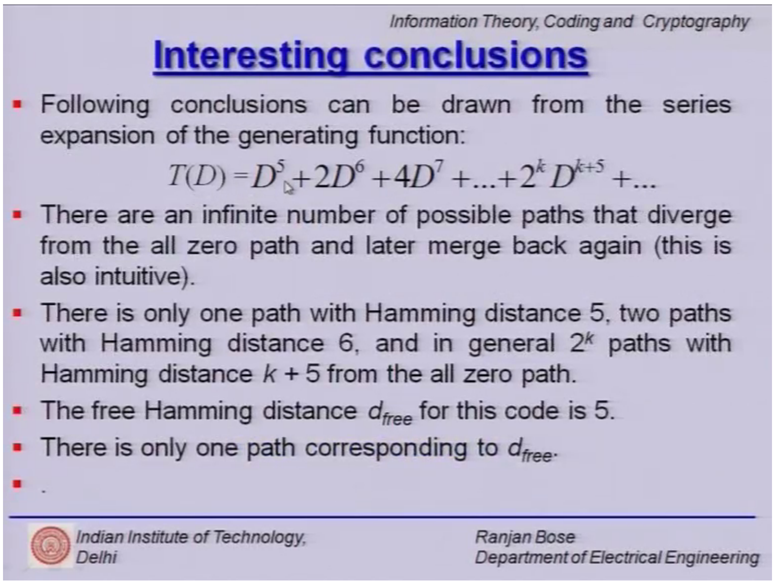
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, if you solve this these sets of state equations we can get T D equal to this D raise for 5 over 1 minus 2 D and, you can have an infinite series expansion for this and you end up with the following series expansion D raise for 5 plus 2 D 6 plus 4 D 7 and so and so, forth in general it can be represented as a summation of a D capital D raised to power small d.

Now, this is very very instructive, first it can be obtained very usually easily using the masons gain formula that we have studied in digital signal processing. So, whether you solve those three equations, or you use the masons gain formula at the end you will obtain, the expression of td in terms of D’s, but the interpretation is very interesting.

(Refer Slide Time: 37:00)



Information Theory, Coding and Cryptography

Interesting conclusions

- Following conclusions can be drawn from the series expansion of the generating function:

$$T(D) = D^5 + 2D^6 + 4D^7 + \ldots + 2^k D^{k+5} + \ldots$$

- There are an infinite number of possible paths that diverge from the all zero path and later merge back again (this is also intuitive).
- There is only one path with Hamming distance 5, two paths with Hamming distance 6, and in general 2^k paths with Hamming distance $k + 5$ from the all zero path.
- The free Hamming distance d_{free} for this code is 5.
- There is only one path corresponding to d_{free}.
- .

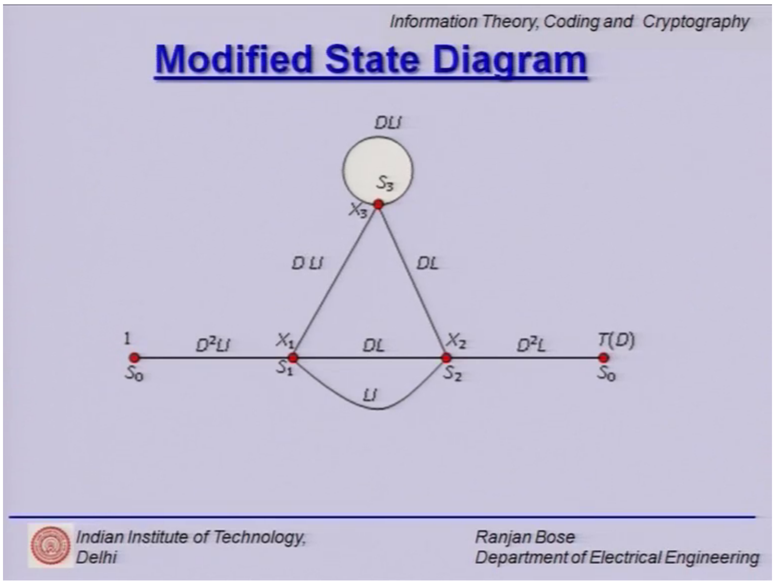
Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, you have this T D equal to D's for 5 plus 2 D's 6 and so on and so, forth. First it shows you that there are infinite terms in this series, pertaining to the infinite possible paths that will diverge from state 0 and merge back to state 0, but there is one path in that set of infinite paths which gives the weight 5 as shown by the exponent 5. And then there 2 paths distinct paths which give weight 6 and, then there four paths which give weight 7 what, do you mean by give weight 7. It means that these 2 paths which diverges from the all 0 paths and merges back, they are toward the hamming distance between them is 7 from the point it diverges and it merges back.

Similarly there 2 raise for k path of the weight k plus 5. So, just by observing the first term in this infinite series I can tell you, that the D free is 5 given by the exponent. So, very easily I can get an answer without worrying about a fact that have I missed out any pair of paths.

(Refer Slide Time: 38:23)



Information Theory, Coding and Cryptography

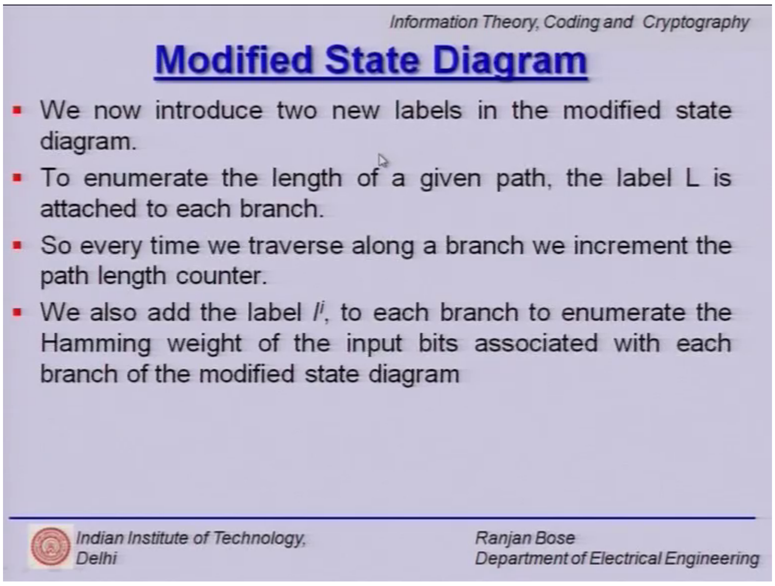
Modified State Diagram

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

Now, we take this concept further and look at this modified state diagram, which we looked earlier, but this time we add not only the weights, but we add two more terms in terms of l and I which will count the hop number of hops.

(Refer Slide Time: 38:45)



Information Theory, Coding and Cryptography

Modified State Diagram

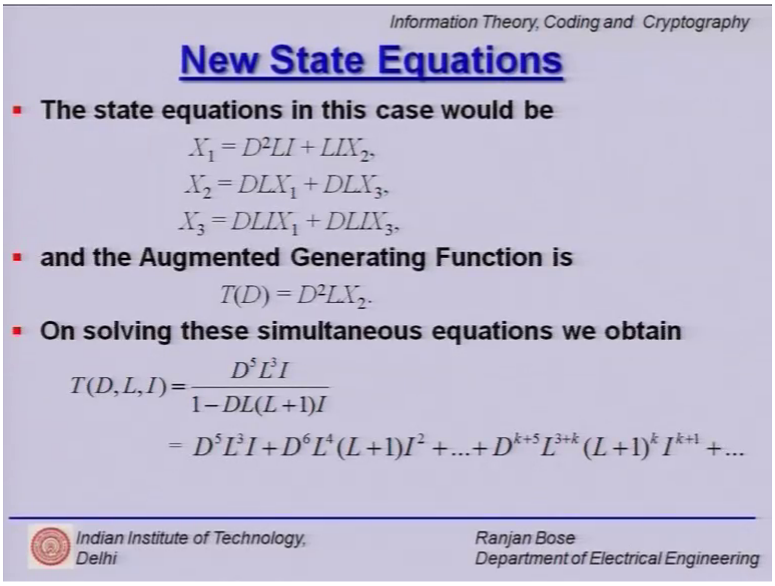
- We now introduce two new labels in the modified state diagram.
- To enumerate the length of a given path, the label L is attached to each branch.
- So every time we traverse along a branch we increment the path length counter.
- We also add the label I^i, to each branch to enumerate the Hamming weight of the input bits associated with each branch of the modified state diagram

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, the two new labels that we add are to enumerate the length of the given path l right. So, what we do is every time it travels along a branch, we increment the path length counter by 1, because the exponent is 1. So, will know how many hops we have gone through and, what we do is we also add the label i capital I raise the bar small i to each branch to enumerate the hamming weight of the input bits associated with it. So, we are talking about the input bit also, why do not we take that counter along as well.

(Refer Slide Time: 39:32)



Information Theory, Coding and Cryptography

New State Equations

- The state equations in this case would be

$$X_1 = D^2LI + LIX_2,$$
$$X_2 = DLX_1 + DLX_3,$$
$$X_3 = DLIX_1 + DLIX_3,$$

- and the Augmented Generating Function is

$$T(D) = D^2LX_2.$$

- On solving these simultaneous equations we obtain

$$T(D,L,I) = \frac{D^5L^3I}{1-DL(L+1)I}$$
$$= D^5L^3I + D^6L^4(L+1)I^2 + \ldots + D^{k+5}L^{3+k}(L+1)^k I^{k+1} + \ldots$$

Indian Institute of Technology, Delhi

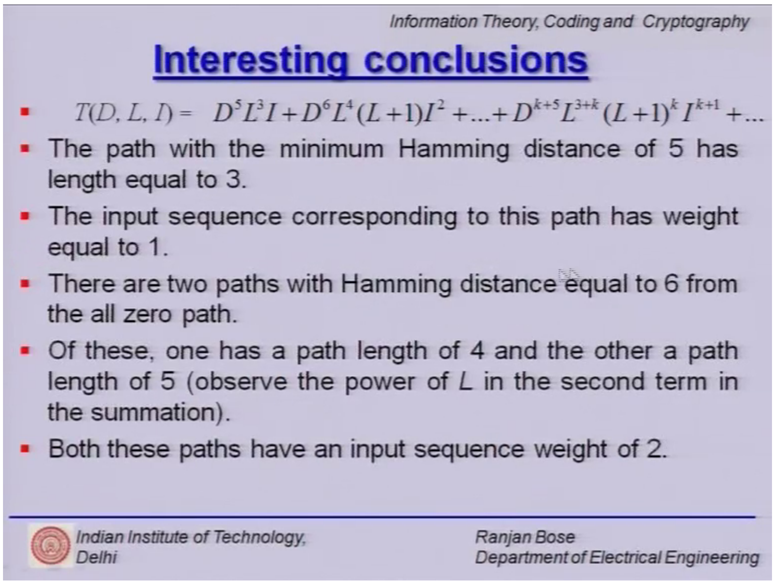
Ranjan Bose
Department of Electrical Engineering

So, if you introduce these two new variables, then you have modified set of state equations given as follows, very similar to what we did last time. And so, this augmented generating function T D is now given by D squared l X 2 as the final 4th equation. We now solve these sets of equations using this dummy variables X 1, X 2 and X 3, we eliminate them and we now have an expression of this augmented generating function T D L I. So, this is a function of D L and I, the D will tell you the hamming weight, l will tell you the branch length and, I will talk about the input of the input weight.

So, now we have another infinite series as expected. So, if you look at the first term which tells us the minimum weight pertaining to the D free. The first exponent tells us that the minimum distance D free is indeed 5, but it also tells us there were three hops ok. So, we went through 3 hops the path length is 3, if you ask me how many hops does the path have which contributes to the weight D equal to 6, we will say that it has got 4. So, there are 2 of 2 such paths 1 path has 5 hops and 1 path has 4 hops earlier this has come out as 2 D raise for 6, 2 times D raise for 6, but in into 2 distinct path.

Now, we now know that these 2 paths are of different lengths, 1 went through 5 hops the other one went only 4 hops. Similarly you have other expressions.

(Refer Slide Time: 41:30)



Information Theory, Coding and Cryptography

Interesting conclusions

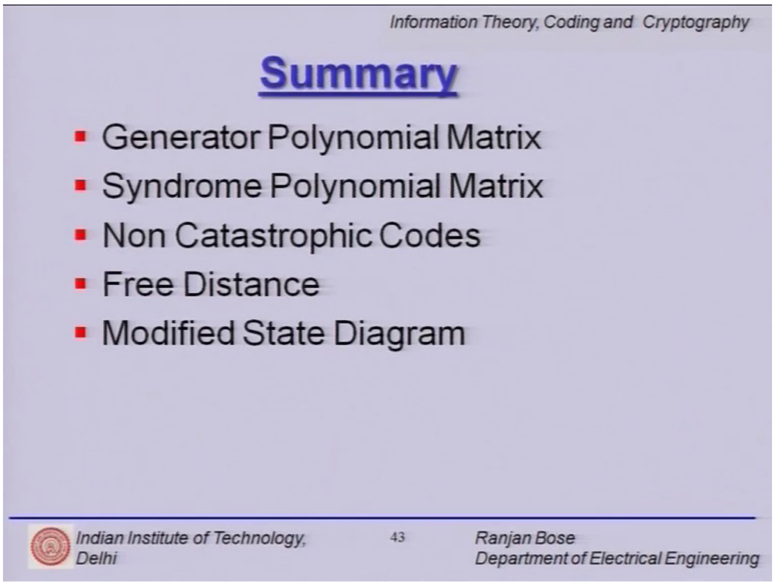
- $T(D, L, I) = D^5 L^3 I + D^6 L^4 (L+1) I^2 + ... + D^{k+5} L^{3+k} (L+1)^k I^{k+1} + ...$
- The path with the minimum Hamming distance of 5 has length equal to 3.
- The input sequence corresponding to this path has weight equal to 1.
- There are two paths with Hamming distance equal to 6 from the all zero path.
- Of these, one has a path length of 4 and the other a path length of 5 (observe the power of *L* in the second term in the summation).
- Both these paths have an input sequence weight of 2.

Indian Institute of Technology, Delhi

Ranjan Bose
Department of Electrical Engineering

So, these are the conclusions we drew the path with the minimum hamming distance of 5 has length equal to 3 and, the input sequence corresponding to this path has weight equal to 1 as given by the exponent I. And, then there are 2 paths with hamming distance equal to 6 from the all 0 paths of length 4 and 5 respectively, but both of these have input sequence of weight equal to 2.

(Refer Slide Time: 41:56)



Information Theory, Coding and Cryptography

Summary

- Generator Polynomial Matrix
- Syndrome Polynomial Matrix
- Non Catastrophic Codes
- Free Distance
- Modified State Diagram

Indian Institute of Technology, Delhi

43

Ranjan Bose
Department of Electrical Engineering

So, now we come to the summary of today's lecture, we have covered some very important points, we started with the generator polynomial matrix and, we moved on to the syndrome polynomial matrix and, we learned how to get the code word polynomial vector and, the syndrome polynomial vector, or matrix whichever way you want to define. Then we talked about both catastrophic and non catastrophic codes and our aim is to design good, non catastrophic codes that do not have this catastrophic error propagation.

We then went on to define the free distance of the code and the D free and how it is linked to the error correcting capability of convolutional codes. Finally, we looked at the modified state diagram, which quickly efficiently and elegantly gives us the D free and it all not only gives you the D free, but the number of hops as well.

With that, we come to the end of today’s lecture.