

An Introduction to Coding Theory
Professor Adrish Banerji
Department of Electrical Engineering
Indian Institute of Technology, Kanpur
Module 07
Lecture Number 29
Problem Solving Session-VI

(Refer Slide Time 00:14)

An introduction to coding theory

Adrish Banerjee

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

Mar. 6, 2017

So in this lecture we will try to solve some problems

(Refer Slide Time 00:16)



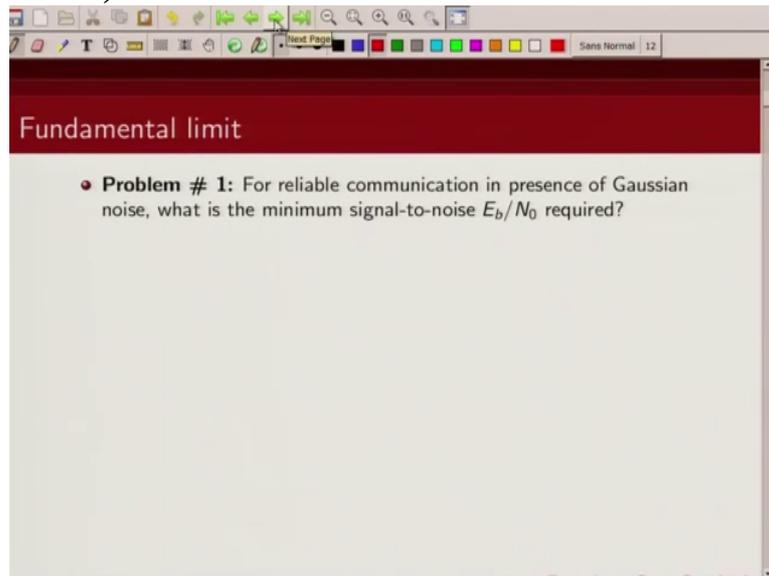
Lecture #16B: Problem solving session-VI

Adrish Banerjee
An introduction to coding theory

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

related to convolutional code in general.

(Refer Slide Time 00:20)



So the first question that we will try to answer is, if you want to do reliable communication in presence of additive white Gaussian noise channel and of course we have infinite bandwidth what is the minimum signal to noise ratio

(Refer Slide Time 00:37)



required?

(Refer Slide Time 00:41)

Fundamental limit

- **Problem # 1:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required?
- **Solution:** Capacity of Gaussian memoryless channel with two-sided noise power spectral density $N_0/2$ and without bandwidth limitation is given by

$$C^\infty = \lim_{W \rightarrow \infty} W \log \left(1 + \frac{S}{N_0 W} \right)$$
$$= \frac{S}{N_0 \ln 2} \text{ bits/s}$$

where W denotes the bandwidth and S is the signaling power.

So for that we first need the expression for capacity of additive white Gaussian noise channel and the capacity of additive white Gaussian noise channel is given by this expression

(Refer Slide Time 00:58)

Fundamental limit

- **Problem # 1:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required?
- **Solution:** Capacity of Gaussian memoryless channel with two-sided noise power spectral density $N_0/2$ and without bandwidth limitation is given by

$$C^\infty = \lim_{W \rightarrow \infty} W \log \left(1 + \frac{S}{N_0 W} \right)$$
$$= \frac{S}{N_0 \ln 2} \text{ bits/s}$$

where W denotes the bandwidth and S is the signaling power.

where w is the bandwidth, s is my signaling power, n naught by 2 is two sided power spectral density. Now, so we are considering when bandwidth is infinite. So this can be written, so this will be $\log 1$ plus when w is infinite this will go to zero so \log of 1 will be zero and 1 by w will also go to zero so zero by zero form. So we will differentiate and we can find out that the capacity when bandwidth is infinite is given by this expression,

(Refer Slide Time 01:44)

Fundamental limit

- **Problem # 1:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required?
- **Solution:** Capacity of Gaussian memoryless channel with two-sided noise power spectral density $N_0/2$ and without bandwidth limitation is given by

$$C^\infty = \lim_{W \rightarrow \infty} W \log \left(1 + \frac{S}{N_0 W} \right)$$
$$= \frac{S}{N_0 \ln 2} \text{ bits/s}$$

where W denotes the bandwidth and S is the signaling power.

s by n naught, natural log of 2 bits per second. So we are interested

(Refer Slide Time 01:54)

Fundamental limit

- **Problem # 1:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required?
- **Solution:** Capacity of Gaussian memoryless channel with two-sided noise power spectral density $N_0/2$ and without bandwidth limitation is given by

$$C^\infty = \lim_{W \rightarrow \infty} W \log \left(1 + \frac{S}{N_0 W} \right)$$
$$= \frac{S}{N_0 \ln 2} \text{ bits/s}$$

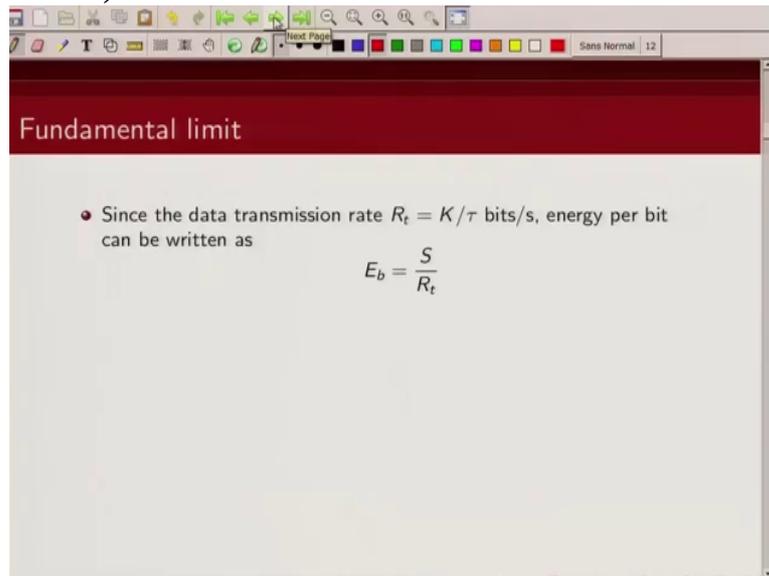
where W denotes the bandwidth and S is the signaling power.

- If we transmit K information bits over τ seconds, where τ is a multiple of T , we have

$$E_b = \frac{S\tau}{K}$$

in transmitting k bits over τ seconds. So if we do that, where τ is the multiple of time period, so if we do that, our energy per bit is given by $s\tau$ by k . This s was my signaling power we are sending over time t , τ and total number of information bits was k . Energy per bit is $s\tau$ divided by k .

(Refer Slide Time 02:30)



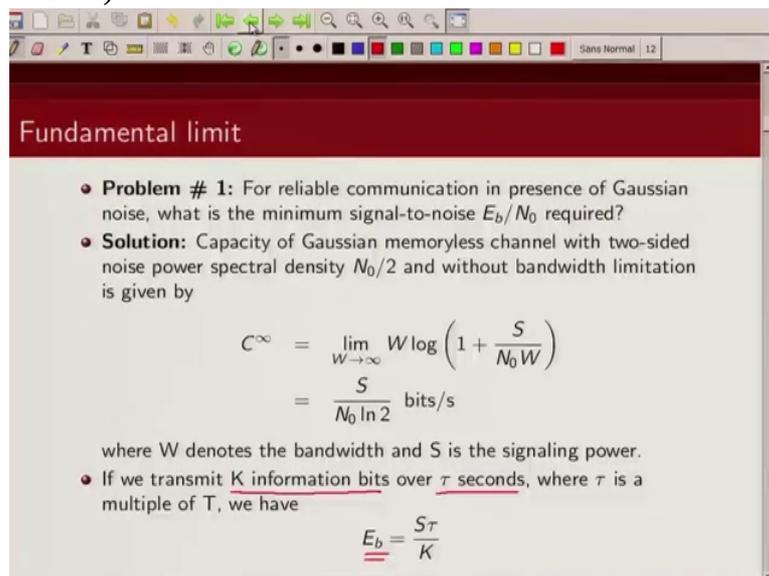
The slide is titled "Fundamental limit" and contains the following text:

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as

$$E_b = \frac{S}{R_t}$$

Now our transmission because you are transmitting k bits over time tau, so our transmission rate is k by tau bits per second. And our energy per bit that we wrote here

(Refer Slide Time 02:46)



The slide is titled "Fundamental limit" and contains the following text:

- **Problem # 1:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required?
- **Solution:** Capacity of Gaussian memoryless channel with two-sided noise power spectral density $N_0/2$ and without bandwidth limitation is given by

$$\begin{aligned} C^\infty &= \lim_{W \rightarrow \infty} W \log \left(1 + \frac{S}{N_0 W} \right) \\ &= \frac{S}{N_0 \ln 2} \text{ bits/s} \end{aligned}$$

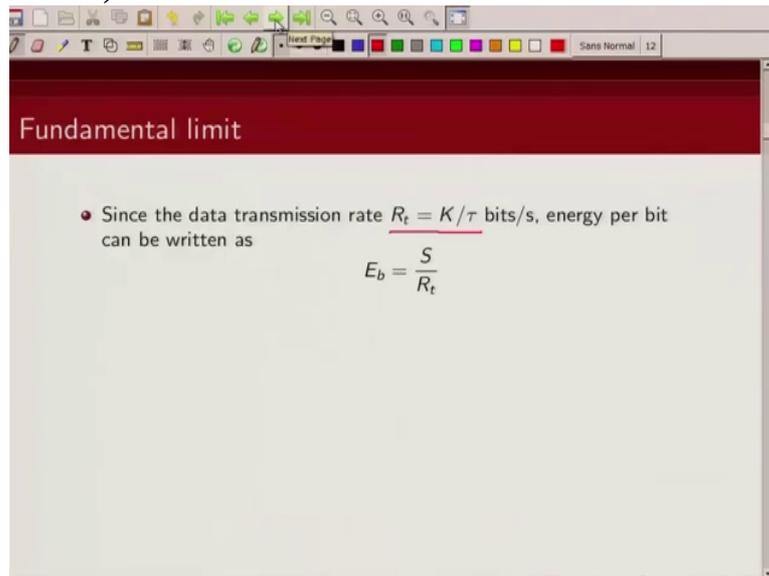
where W denotes the bandwidth and S is the signaling power.

- If we transmit K information bits over τ seconds, where τ is a multiple of T, we have

$$\underline{E_b} = \frac{S\tau}{K}$$

is basically s tau by k. And

(Refer Slide Time 02:51)



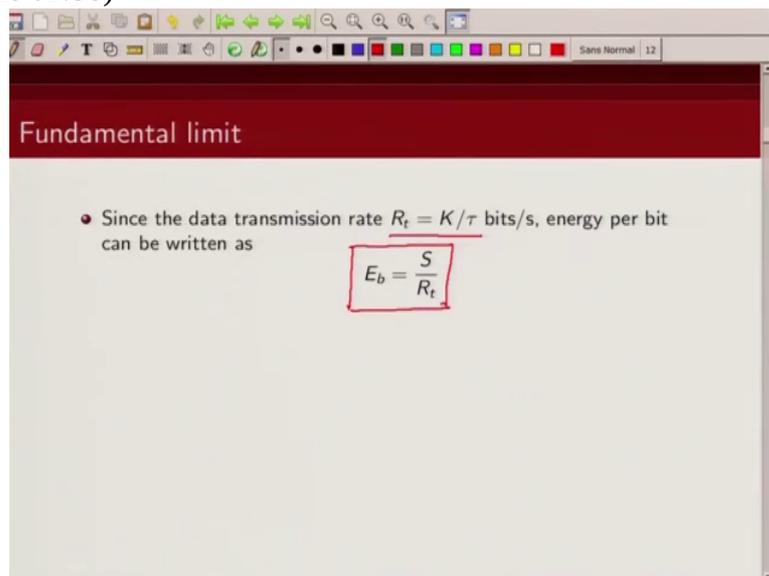
Fundamental limit

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as

$$E_b = \frac{S}{R_t}$$

k by tau is R_t . So we can write energy per bit in terms of signaling power

(Refer Slide Time 02:58)



Fundamental limit

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as

$$E_b = \frac{S}{R_t}$$

and transmission rate R_t . So

(Refer Slide Time 03:03)

The screenshot shows a presentation slide with a red header containing the text "Fundamental limit". Below the header, there is a bulleted list with two items. The first item states: "Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as". To the right of this text is the equation
$$E_b = \frac{S}{R_t}$$
. The second item in the list states: "Thus we have". To the right of this text is the equation
$$\frac{C^\infty}{R_t} = \frac{E_b}{N_0 \ln 2}$$
. The slide also features a standard software toolbar at the top with various icons and a "Next Page" button.

if we divide our

(Refer Slide Time 03:05)

The screenshot shows a presentation slide with a red header containing the text "Fundamental limit". Below the header, there is a bulleted list with one item. The item states: "Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as". To the right of this text, the equation
$$E_b = \frac{S}{R_t}$$
 is displayed and enclosed in a red rectangular box. The slide also features a standard software toolbar at the top with various icons and a "Previous Page" button.

expression

(Refer Slide Time 03:07)

The slide is titled "Fundamental limit" and contains the following content:

- **Problem # 1:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required?
- **Solution:** Capacity of Gaussian memoryless channel with two-sided noise power spectral density $N_0/2$ and without bandwidth limitation is given by

$$C^\infty = \lim_{W \rightarrow \infty} W \log \left(1 + \frac{S}{N_0 W} \right)$$
$$= \frac{S}{N_0 \ln 2} \text{ bits/s}$$

where W denotes the bandwidth and S is the signaling power.

- If we transmit K information bits over τ seconds, where τ is a multiple of T , we have

$$\underline{E_b} = \frac{S\tau}{K}$$

for channel capacity by R_t what we

(Refer Slide Time 03:11)

The slide is titled "Fundamental limit" and contains the following content:

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as

$$E_b = \frac{S}{R_t}$$

get is this

(Refer Slide Time 03:13)

Fundamental limit

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as
$$E_b = \frac{S}{R_t}$$
- Thus we have
$$\frac{C^\infty}{R_t} = \frac{E_b}{N_0 \ln 2}$$

expression.

(Refer Slide Time 03:18)

Fundamental limit

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as
$$E_b = \frac{S}{R_t}$$
- Thus we have
$$\frac{C^\infty}{R_t} = \frac{E_b}{N_0 \ln 2}$$

Now we know from Shannon Noisy channel coding theorem that as long as transmission rate is less than channel capacity we can reliably communicate over the communication channel. So we want R_t to be less than this channel capacity.

(Refer Slide Time 03:40)

The slide is titled "Fundamental limit" and contains the following text and equations:

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as
$$E_b = \frac{S}{R_t}$$
- Thus we have
$$\frac{C^\infty}{R_t} = \frac{E_b}{N_0 \ln 2}$$
- For reliable communication, we must have $R_t < C^\infty$. Thus we have
$$\frac{E_b}{N_0} > \ln 2 = 0.69 = -1.6 \text{ dB}$$

So if we want this to hold then we get a condition on signal to noise ratio. So we get this following condition on

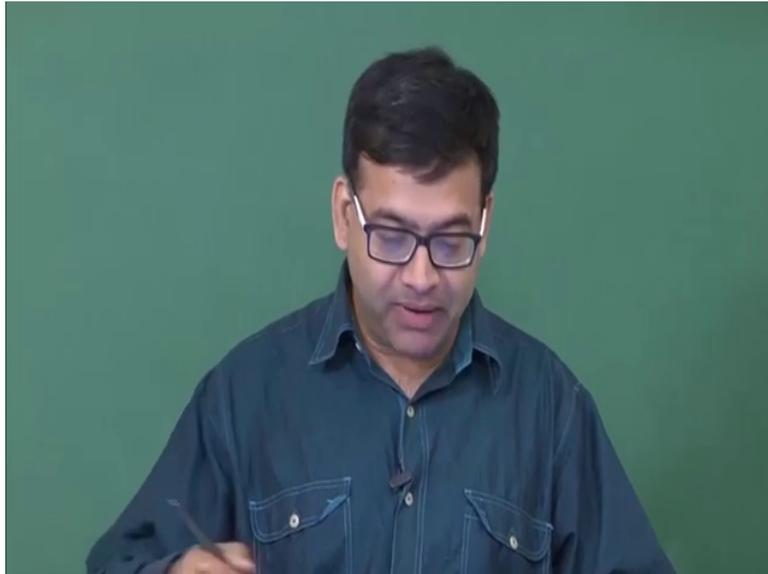
(Refer Slide Time 03:53)

The slide is titled "Fundamental limit" and contains the following text and equations:

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as
$$E_b = \frac{S}{R_t}$$
- Thus we have
$$\frac{C^\infty}{R_t} = \frac{E_b}{N_0 \ln 2}$$
- For reliable communication, we must have $R_t < C^\infty$. Thus we have
$$\frac{E_b}{N_0} > \ln 2 = 0.69 = -1.6 \text{ dB}$$

energy per bit. So s n R per information bit, we get this condition that s n R per information bit should be greater than 1, minus 1 point 6 D B. This is for the case when we have infinite

(Refer Slide Time 04:12)



bandwidth and our code rate

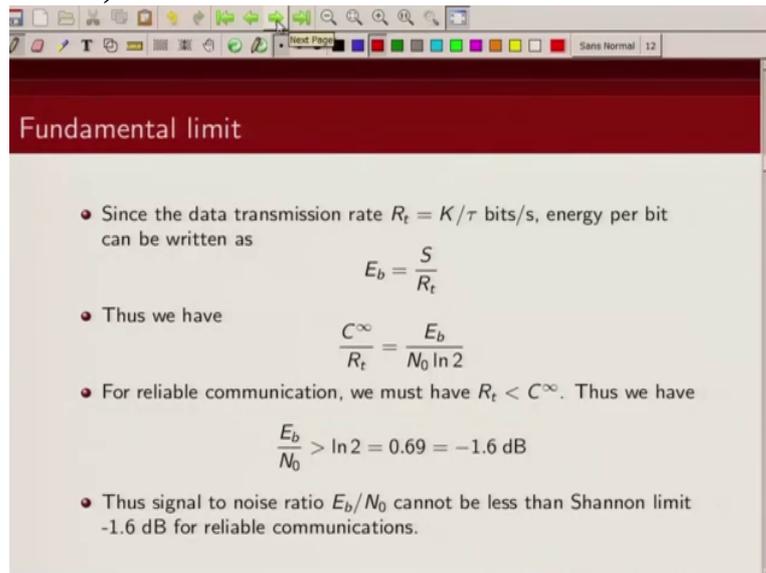
(Refer Slide Time 04:14)

Fundamental limit

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as
$$E_b = \frac{S}{R_t}$$
- Thus we have
$$\frac{C^\infty}{R_t} = \frac{E_b}{N_0 \ln 2}$$
- For reliable communication, we must have $R_t < C^\infty$. Thus we have
$$\frac{E_b}{N_0} > \ln 2 = 0.69 = -1.6 \text{ dB}$$

can actually go to zero. So

(Refer Slide Time 04:17)



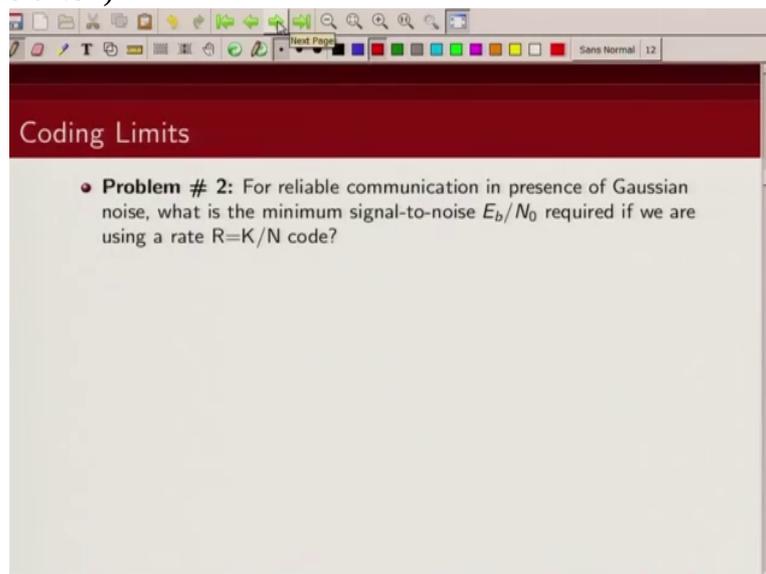
The slide is titled "Fundamental limit" and contains the following text and equations:

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as
$$E_b = \frac{S}{R_t}$$
- Thus we have
$$\frac{C^\infty}{R_t} = \frac{E_b}{N_0 \ln 2}$$
- For reliable communication, we must have $R_t < C^\infty$. Thus we have
$$\frac{E_b}{N_0} > \ln 2 = 0.69 = -1.6 \text{ dB}$$
- Thus signal to noise ratio E_b/N_0 cannot be less than Shannon limit -1.6 dB for reliable communications.

signal to noise ratio then cannot be less than this limit which is minus 1 point 6 D B.

OK let's look at next problem. So

(Refer Slide Time 04:34)

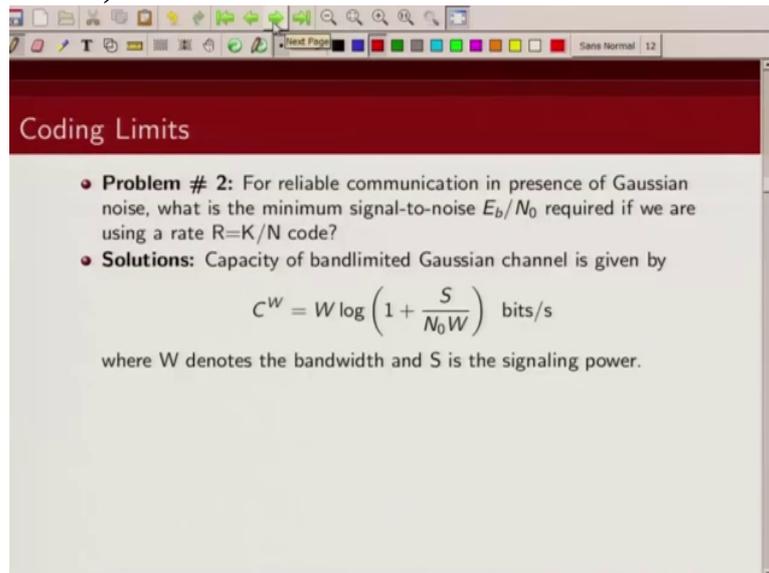


The slide is titled "Coding Limits" and contains the following text:

- **Problem # 2:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required if we are using a rate $R=K/N$ code?

we are interested in reliable communication over additive white Gaussian noise channel and we are transmitting using a rate k by n code. So what is the affect of the

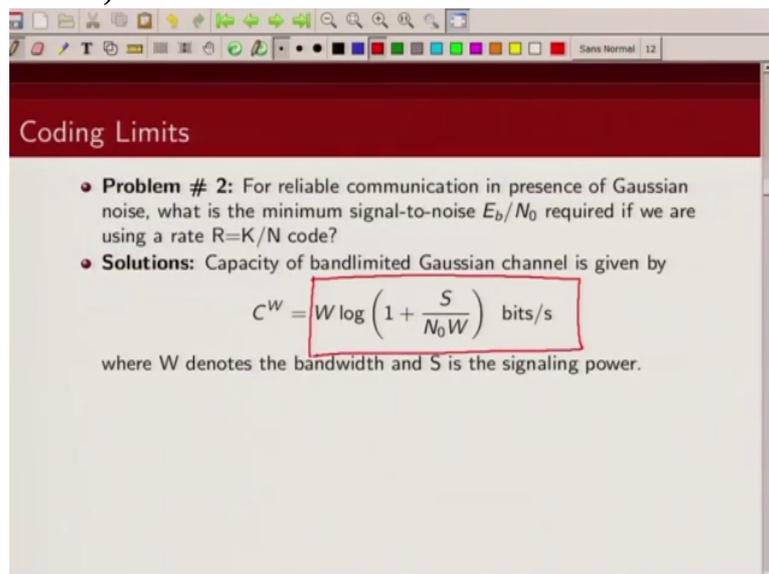
(Refer Slide Time 04:49)



The screenshot shows a presentation slide with a dark red header containing the text "Coding Limits". Below the header, there are two bullet points. The first bullet point is "Problem # 2: For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required if we are using a rate $R=K/N$ code?". The second bullet point is "Solutions: Capacity of bandlimited Gaussian channel is given by" followed by the equation $C^W = W \log \left(1 + \frac{S}{N_0 W} \right)$ bits/s. Below the equation, it says "where W denotes the bandwidth and S is the signaling power." The slide is displayed in a software window with a standard toolbar at the top.

bandwidth w ? Now we know for a band limited channel, the capacity is given by this expression.

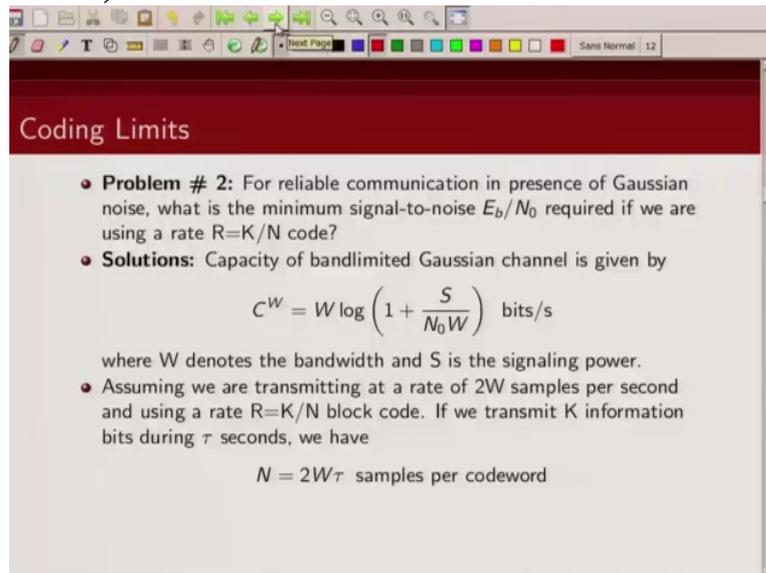
(Refer Slide Time 05:00)



This screenshot is identical to the one above, showing the same "Coding Limits" slide. However, in this version, the equation $C^W = W \log \left(1 + \frac{S}{N_0 W} \right)$ bits/s is enclosed in a red rectangular box, highlighting it.

So

(Refer Slide Time 05:03)

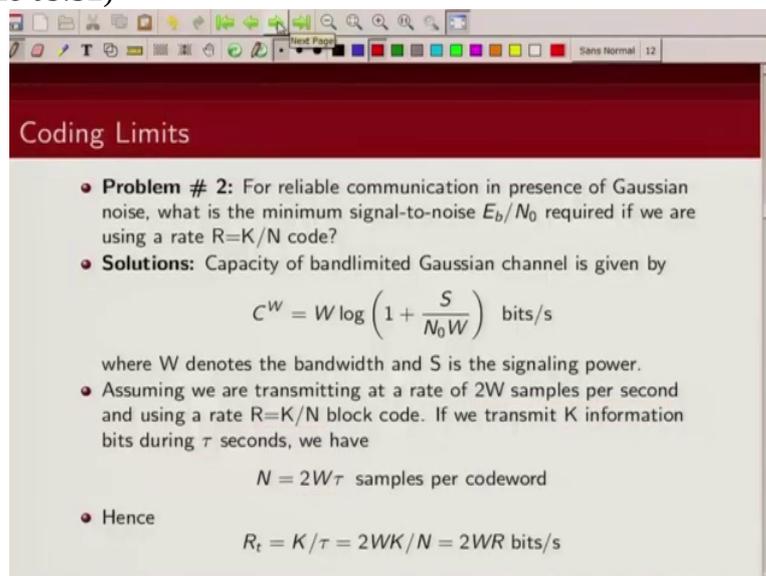


The slide is titled "Coding Limits" and contains the following text:

- **Problem # 2:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required if we are using a rate $R=K/N$ code?
- **Solutions:** Capacity of bandlimited Gaussian channel is given by
$$C^W = W \log \left(1 + \frac{S}{N_0 W} \right) \text{ bits/s}$$
where W denotes the bandwidth and S is the signaling power.
- Assuming we are transmitting at a rate of $2W$ samples per second and using a rate $R=K/N$ block code. If we transmit K information bits during τ seconds, we have
$$N = 2W\tau \text{ samples per codeword}$$

let's say we are transmitting at Nyquist rate. So we are transmitting at $2w$ samples per second. And we are using a rate k by n code. So if we transmit k information bits during time τ then number of samples per codeword we are sending is $2w\tau$.

(Refer Slide Time 05:32)

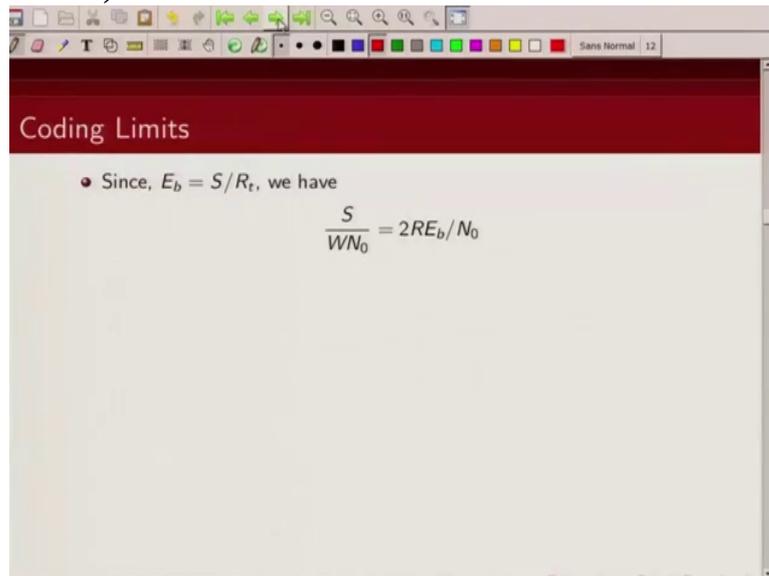


The slide is titled "Coding Limits" and contains the following text:

- **Problem # 2:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required if we are using a rate $R=K/N$ code?
- **Solutions:** Capacity of bandlimited Gaussian channel is given by
$$C^W = W \log \left(1 + \frac{S}{N_0 W} \right) \text{ bits/s}$$
where W denotes the bandwidth and S is the signaling power.
- Assuming we are transmitting at a rate of $2W$ samples per second and using a rate $R=K/N$ block code. If we transmit K information bits during τ seconds, we have
$$N = 2W\tau \text{ samples per codeword}$$
- Hence
$$R_t = K/\tau = 2WK/N = 2WR \text{ bits/s}$$

Now our transmission rates is k information bits over τ time. So our transmission rate is k by τ and this we can write in terms of these samples per codeword n which is given by this expression and k by n is nothing but our code rate. So this transmission rate is given by $2wR$ where R is my code rate. So this many bits per second is my transmission rate. Now

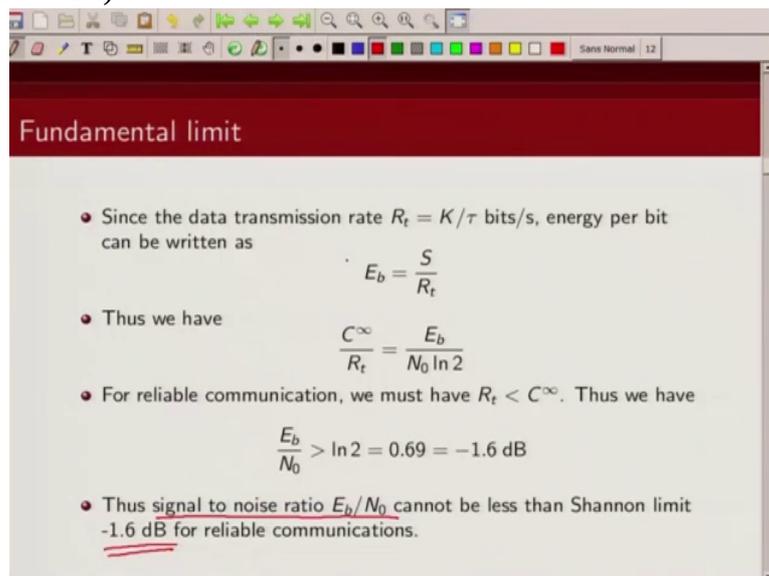
(Refer Slide Time 06:09)



The slide is titled "Coding Limits" in a dark red header. Below the header, there is a bullet point: "• Since, $E_b = S/R_t$, we have". Below this, the equation $\frac{S}{WN_0} = 2RE_b/N_0$ is displayed in a large font.

we know that energy per bit is given by signaling power by transmission rate, this we have done in the last example.

(Refer Slide Time 06:25)



The slide is titled "Fundamental limit" in a dark red header. It contains several bullet points and equations:

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as
$$E_b = \frac{S}{R_t}$$
- Thus we have
$$\frac{C^\infty}{R_t} = \frac{E_b}{N_0 \ln 2}$$
- For reliable communication, we must have $R_t < C^\infty$. Thus we have
$$\frac{E_b}{N_0} > \ln 2 = 0.69 = -1.6 \text{ dB}$$
- Thus signal to noise ratio E_b/N_0 cannot be less than Shannon limit -1.6 dB for reliable communications.

We had this, right

(Refer Slide Time 06:28)

The slide is titled "Fundamental limit" and contains the following content:

- Since the data transmission rate $R_t = K/\tau$ bits/s, energy per bit can be written as
$$E_b = \frac{S}{R_t}$$
- Thus we have
$$\frac{C^\infty}{R_t} = \frac{E_b}{N_0 \ln 2}$$
- For reliable communication, we must have $R_t < C^\infty$. Thus we have
$$\frac{E_b}{N_0} > \ln 2 = 0.69 = -1.6 \text{ dB}$$
- Thus signal to noise ratio E_b/N_0 cannot be less than Shannon limit -1.6 dB for reliable communications.

so using this, then

(Refer Slide Time 06:33)

The slide is titled "Coding Limits" and contains the following content:

- Since, $E_b = S/R_t$, we have
$$\frac{S}{WN_0} = 2RE_b/N_0$$
- For reliable communications, we must have $R_t < C^W$, thus
$$R_t = 2WR < W \log \left(1 + \frac{2RE_b}{N_0} \right)$$

so we do that we can write s by w n naught in terms of

(Refer Slide Time 06:41)

The slide is titled "Coding Limits" and contains the following content:

- Since, $E_b = S/R_t$, we have

$$\frac{S}{WN_0} = 2RE_b/N_0$$

- For reliable communications, we must have $R_t < C^W$, thus

$$R_t = 2WR < W \log \left(1 + \frac{2RE_b}{N_0} \right)$$

E_b by N_0 , this is equal to 2 times code rate by, into S/N per information bit. So in this expression of

(Refer Slide Time 06:55)

The slide is titled "Coding Limits" and contains the following content:

- **Problem # 2:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required if we are using a rate $R=K/N$ code?
- **Solutions:** Capacity of bandlimited Gaussian channel is given by

$$C^W = W \log \left(1 + \frac{S}{N_0W} \right) \text{ bits/s}$$

where W denotes the bandwidth and S is the signaling power.

- Assuming we are transmitting at a rate of $2W$ samples per second and using a rate $R=K/N$ block code. If we transmit K information bits during τ seconds, we have

$$N = 2W\tau \text{ samples per codeword}$$

- Hence

$$R_t = K/\tau = \underline{\underline{2WK/N}} = \underline{\underline{2WR}} \text{ bits/s}$$

capacity of band limited channel, we can replace this

(Refer Slide Time 07:01)

The slide is titled "Coding Limits" and contains the following text:

- **Problem # 2:** For reliable communication in presence of Gaussian noise, what is the minimum signal-to-noise E_b/N_0 required if we are using a rate $R=K/N$ code?
- **Solutions:** Capacity of bandlimited Gaussian channel is given by
$$C^W = W \log \left(1 + \frac{S}{N_0 W} \right) \text{ bits/s}$$
where W denotes the bandwidth and S is the signaling power.
- Assuming we are transmitting at a rate of $2W$ samples per second and using a rate $R=K/N$ block code. If we transmit K information bits during τ seconds, we have
$$N = 2W\tau \text{ samples per codeword}$$
- Hence
$$R_t = K/\tau = \underline{2WK/N} = \underline{2WR} \text{ bits/s}$$

expression by

(Refer Slide Time 07:03)

The slide is titled "Coding Limits" and contains the following text:

- Since, $\underline{E_b = S/R_t}$, we have
$$\frac{S}{WN_0} = 2RE_b/N_0$$

this, fine? Now

(Refer Slide Time 07:11)

The screenshot shows a presentation slide with a red header titled "Coding Limits". The slide contains the following text:

- Since, $E_b = S/R_t$, we have

$$\frac{S}{WN_0} = 2RE_b/N_0$$

- For reliable communications, we must have $R_t < C^W$, thus

$$R_t = 2WR < W \log \left(1 + \frac{2RE_b}{N_0} \right)$$

for reliable communication we know our transmission rate should be less than channel capacity. And what's channel capacity? That's for the band limited channel; it is $w \log$ of 1 plus s by $2 N$ naught which is basically given by this expression. So transmission rate which is 2 times $w R$ should be less than $w \log$ of 1 plus $2 R E_b$ by N naught.

(Refer Slide Time 07:43)

The screenshot shows a presentation slide with a red header titled "Coding Limits". The slide contains the following text:

- Since, $E_b = S/R_t$, we have

$$\frac{S}{WN_0} = 2RE_b/N_0$$

- For reliable communications, we must have $R_t < C^W$, thus

$$R_t = 2WR < W \log \left(1 + \frac{2RE_b}{N_0} \right)$$

- We can write equivalently

$$E_b/N_0 > \frac{2^{2R} - 1}{2R}$$

So from here then we can write down the expression for minimum s n R information bit, energy per information bit by noise power density so s n R per information bit we can write it as, this should be greater than equal to 2 raised to power 2 R minus 1 divided by 2 R . Now this right hand side term is an increasing function of R .

(Refer Slide Time 08:16)

Coding Limits

- Since, $E_b = S/R_t$, we have
$$\frac{S}{WN_0} = 2RE_b/N_0$$
- For reliable communications, we must have $R_t < C^W$, thus
$$R_t = 2WR < W \log \left(1 + \frac{2RE_b}{N_0} \right)$$
- We can write equivalently
$$\underline{E_b/N_0} > \frac{2^{2R} - 1}{2R}$$

of R. So

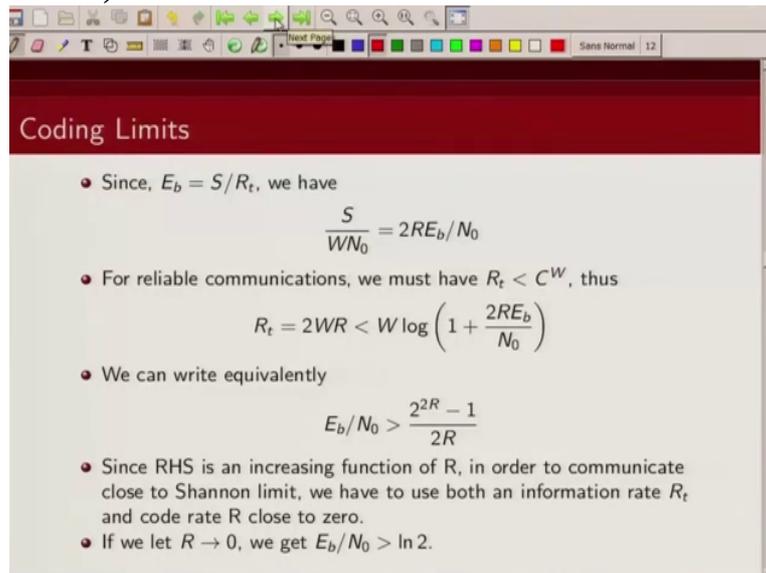
Refer Slide Time 08:17)

Coding Limits

- Since, $E_b = S/R_t$, we have
$$\frac{S}{WN_0} = 2RE_b/N_0$$
- For reliable communications, we must have $R_t < C^W$, thus
$$R_t = 2WR < W \log \left(1 + \frac{2RE_b}{N_0} \right)$$
- We can write equivalently
$$E_b/N_0 > \frac{2^{2R} - 1}{2R}$$
- Since RHS is an increasing function of R, in order to communicate close to Shannon limit, we have to use both an information rate R_t and code rate R close to zero.

So if you want to communicate close to the channel limit then your information rate R as well as code rate R should be close to zero. In fact if you

(Refer Slide Time 08:33)

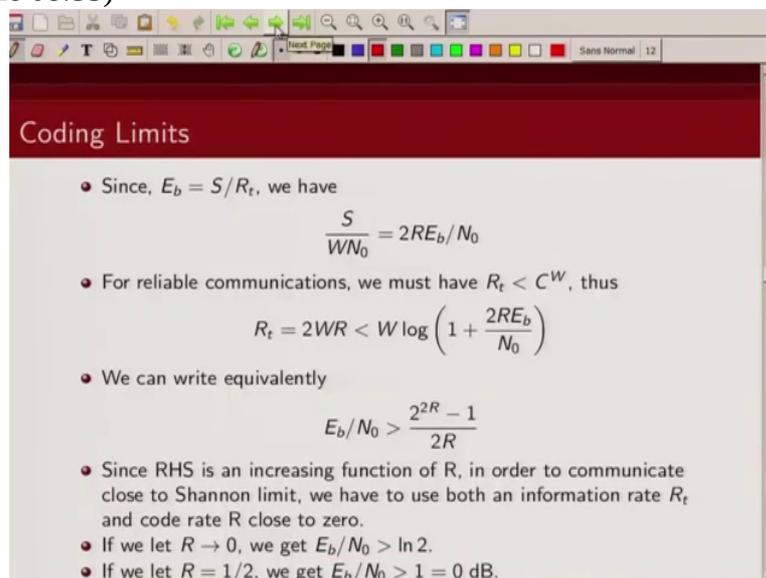


The slide is titled "Coding Limits" and contains the following content:

- Since, $E_b = S/R_t$, we have
$$\frac{S}{WN_0} = 2RE_b/N_0$$
- For reliable communications, we must have $R_t < C^W$, thus
$$R_t = 2WR < W \log \left(1 + \frac{2RE_b}{N_0} \right)$$
- We can write equivalently
$$E_b/N_0 > \frac{2^{2R} - 1}{2R}$$
- Since RHS is an increasing function of R, in order to communicate close to Shannon limit, we have to use both an information rate R_t and code rate R close to zero.
- If we let $R \rightarrow 0$, we get $E_b/N_0 > \ln 2$.

if you let R go to zero, what you will see is you will get the limit that we had just talked about in the previous problem which was minus 1 point 6 d B and let's plug in some practical values of R. Let's say R is half,

(Refer Slide Time 08:53)

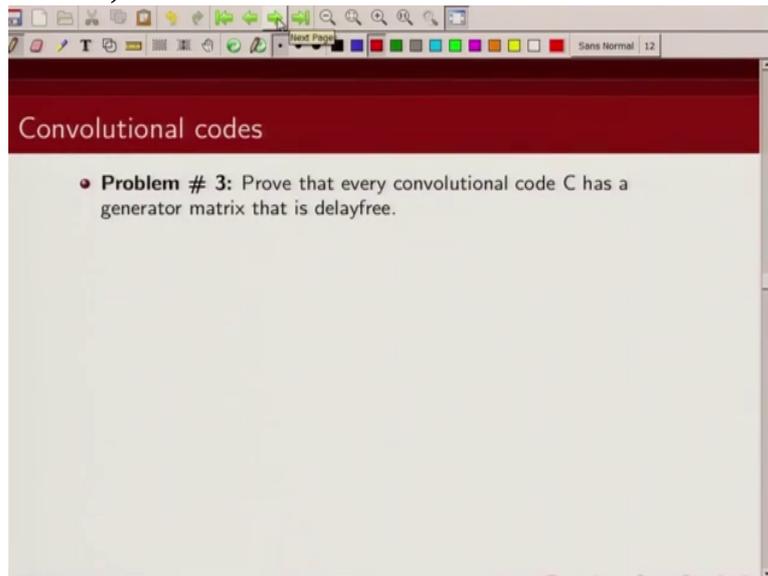


The slide is titled "Coding Limits" and contains the following content:

- Since, $E_b = S/R_t$, we have
$$\frac{S}{WN_0} = 2RE_b/N_0$$
- For reliable communications, we must have $R_t < C^W$, thus
$$R_t = 2WR < W \log \left(1 + \frac{2RE_b}{N_0} \right)$$
- We can write equivalently
$$E_b/N_0 > \frac{2^{2R} - 1}{2R}$$
- Since RHS is an increasing function of R, in order to communicate close to Shannon limit, we have to use both an information rate R_t and code rate R close to zero.
- If we let $R \rightarrow 0$, we get $E_b/N_0 > \ln 2$.
- If we let $R = 1/2$, we get $E_b/N_0 > 1 = 0 \text{ dB}$.

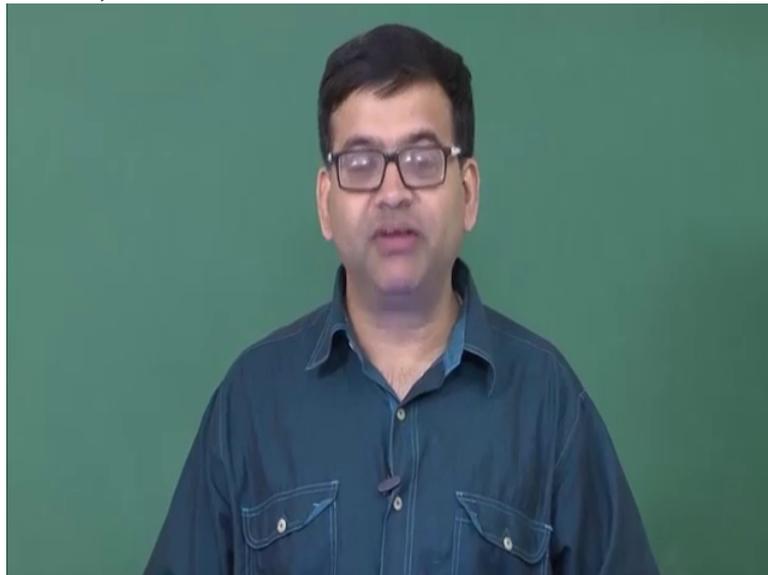
we put value of R to be half then e b by n naught should be more than 0 d B. So you can see for any rate which is away from zero then this minimum s n R required for transmission is also more, Ok.

(Refer Slide Time 09:16)



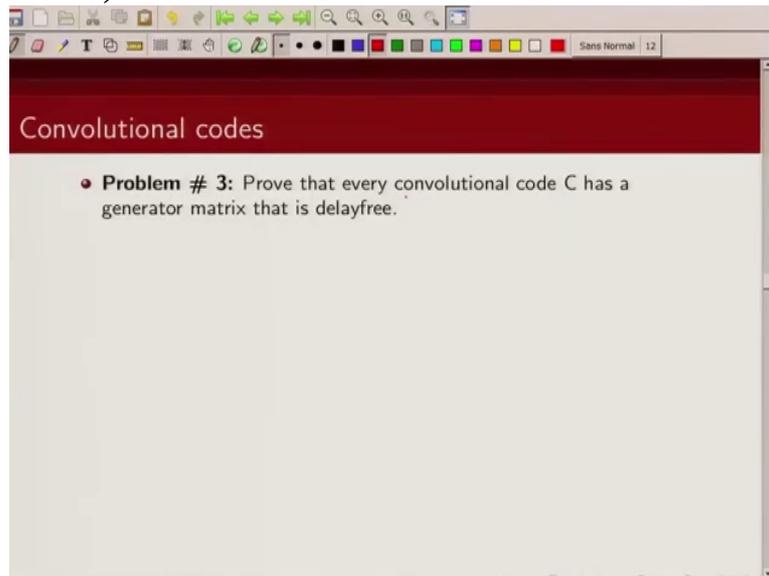
Now the next question that we

(Refer Slide Time 09:19)



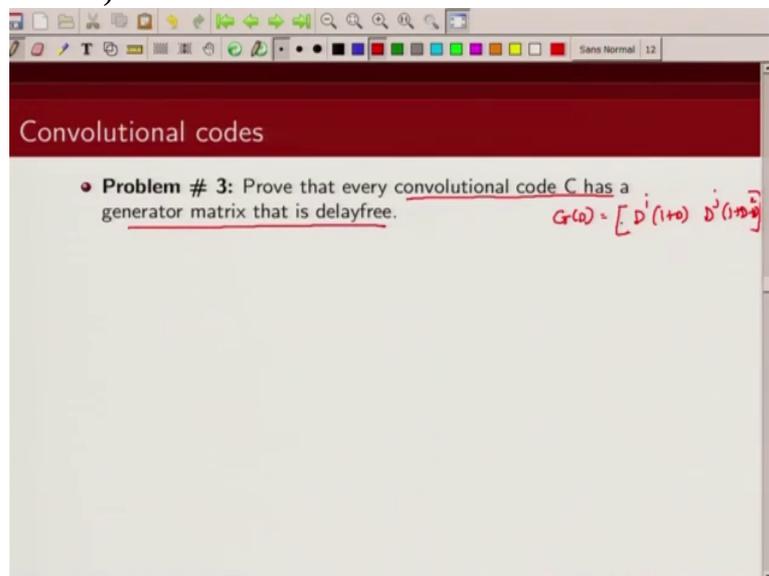
will solve is to prove that

(Refer Slide Time 09:22)



for every convolutional code has a generator matrix which is delay free. Now delay free meaning basically we, if we have a generator matrix let's say of the form D^i and then something like $1 + D$ and some D^j , $1 + D + D^2$ or something like that, we can always write

(Refer Slide Time 09:48)



equivalent generator matrix which will be free of these, like delay terms. I will talk about that.

(Refer Slide Time 09:56)

Convolutional codes

- **Problem # 3:** Prove that every convolutional code C has a generator matrix that is delayfree.
- **Solution:** Let $G(D)$ be any generator matrix for C . The nonzero entries of $G(D)$ can be written as

$$g_{ij}(D) = D^{s_{ij}} f_{ij}(D) / q_{ij}(D)$$
 where s_{ij} is an integer such that

$$f_{ij}(0) = q_{ij}(0) = 1, 1 \leq i \leq k, 1 \leq j \leq n.$$

So let's say we have a generator matrix G of D , Ok. And its non zero entries can be written in this particular form. So there is some delay term, we are calling it D s t raised to power s i j. And then we have this rational function. We have this rational function which is f i j D by q i j D . And f i j D and q i j D is of the form 1 plus some f naught D plus dah dah dah and this is one plus q naught D plus something something,

(Refer Slide Time 10:42)

Convolutional codes

- **Problem # 3:** Prove that every convolutional code C has a generator matrix that is delayfree.
- **Solution:** Let $G(D)$ be any generator matrix for C . The nonzero entries of $G(D)$ can be written as

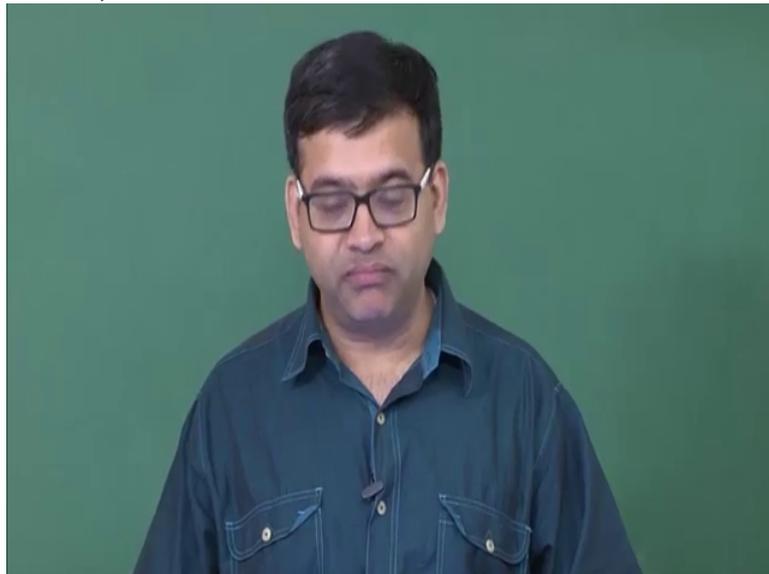
$$g_{ij}(D) = D^{s_{ij}} \left(\frac{f_{ij}(D)}{q_{ij}(D)} \right)$$
 where s_{ij} is an integer such that

$$f_{ij}(0) = q_{ij}(0) = 1, 1 \leq i \leq k, 1 \leq j \leq n.$$

Handwritten note: $\frac{f_{ij}(D)}{q_{ij}(D)} = \frac{1 + f_0 D + \dots}{1 + q_0 D + \dots}$

Ok. So that's what I meant f 0 and q y j zero is 1. So any entry in the generator matrix can be written in this form,

(Refer Slide Time 10:54)



Ok. Now

(Refer Slide Time 10:58)

The image shows a screenshot of a presentation slide. At the top, there is a red header bar with the text "Convolutional codes". Below the header, there is a list of bullet points. The first bullet point is "Problem # 3: Prove that every convolutional code C has a generator matrix that is delayfree." The second bullet point is "Solution: Let $G(D)$ be any generator matrix for C. The nonzero entries of $G(D)$ can be written as". To the right of this text, there is a handwritten equation: $\frac{f_{ij}(D)}{q_{ij}(D)} = \frac{1 + f_0 D + \dots}{1 + q_0 D + \dots}$. Below the text, there is a boxed equation: $g_{ij}(D) = D^{s_{ij}} \frac{f_{ij}(D)}{q_{ij}(D)}$. Below this equation, there is text: "where s_{ij} is an integer such that $f_{ij}(0) = q_{ij}(0) = 1, 1 \leq i \leq k, 1 \leq j \leq n$ ".

Convolutional codes

- **Problem # 3:** Prove that every convolutional code C has a generator matrix that is delayfree.
- **Solution:** Let $G(D)$ be any generator matrix for C. The nonzero entries of $G(D)$ can be written as

$$g_{ij}(D) = D^{s_{ij}} \frac{f_{ij}(D)}{q_{ij}(D)}$$

where s_{ij} is an integer such that $f_{ij}(0) = q_{ij}(0) = 1, 1 \leq i \leq k, 1 \leq j \leq n$.

(Refer Slide Time 10:59)

Convolutional codes

- **Problem # 3:** Prove that every convolutional code C has a generator matrix that is delayfree.
- **Solution:** Let $G(D)$ be any generator matrix for C . The nonzero entries of $G(D)$ can be written as

$$g_{ij}(D) = D^{s_{ij}} f_{ij}(D)/q_{ij}(D)$$
 where s_{ij} is an integer such that $f_{ij}(0) = q_{ij}(0) = 1, 1 \leq i \leq k, 1 \leq j \leq n$.
- The number s_{ij} is the delay of the sequence

$$g_{ij}(D) = D^{s_{ij}} f_{ij}(D)/q_{ij}(D) = D^{s_{ij}} + g_{s_{ij}+1} D^{s_{ij}+1} + \dots$$

what is this term s_{ij} ? Essentially it's a delay term. So if we have some term like s_{ij} so what you will get is terms of this particular form. Now

(Refer Slide Time 11:12)

Convolutional codes

- **Problem # 3:** Prove that every convolutional code C has a generator matrix that is delayfree.
- **Solution:** Let $G(D)$ be any generator matrix for C . The nonzero entries of $G(D)$ can be written as

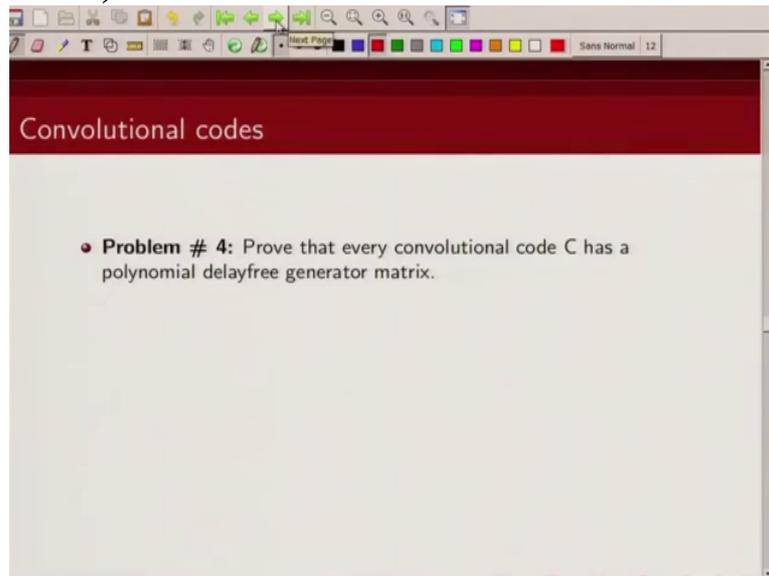
$$g_{ij}(D) = D^{s_{ij}} f_{ij}(D)/q_{ij}(D)$$
 where s_{ij} is an integer such that $f_{ij}(0) = q_{ij}(0) = 1, 1 \leq i \leq k, 1 \leq j \leq n$.
- The number s_{ij} is the delay of the sequence

$$g_{ij}(D) = D^{s_{ij}} f_{ij}(D)/q_{ij}(D) = D^{s_{ij}} + g_{s_{ij}+1} D^{s_{ij}+1} + \dots$$
- Let $s = \min_{i,j} \{s_{ij}\}$, then

$$G'(D) = D^{-s} G(D)$$
 is both delayfree and realizable and both $G(D)$ and $G'(D)$ generate the same convolutional code.

if we consider s which is the minimum s_{ij} over all i and j , then we can find an equivalent generator matrix which would be G' of D which we can write as $G' = D^{-s} G$ and this G' will be delay free and it is of course realizable and it also generates the same set of codewords. So whenever we have a generator matrix which has a delay term we can always find an equivalent generator matrix which is delay free.

(Refer Slide Time 11:55)



Next question, prove that every convolutional code has a polynomial delay free generator matrix. So for any convolutional code,

(Refer Slide Time 12:09)



we have an equivalent polynomial delay free generator matrix.

(Refer Slide Time 12:15)

The screenshot shows a presentation slide with a dark red header containing the text "Convolutional codes". Below the header, there is a list of two bullet points. The first bullet point is "Problem # 4: Prove that every convolutional code C has a polynomial delayfree generator matrix." The second bullet point is "Solutions: Let G(D) be any realizable and delayfree generator for C and let q(D) be the least common multiple of all the denominators of the nonzero entries of G(D)."

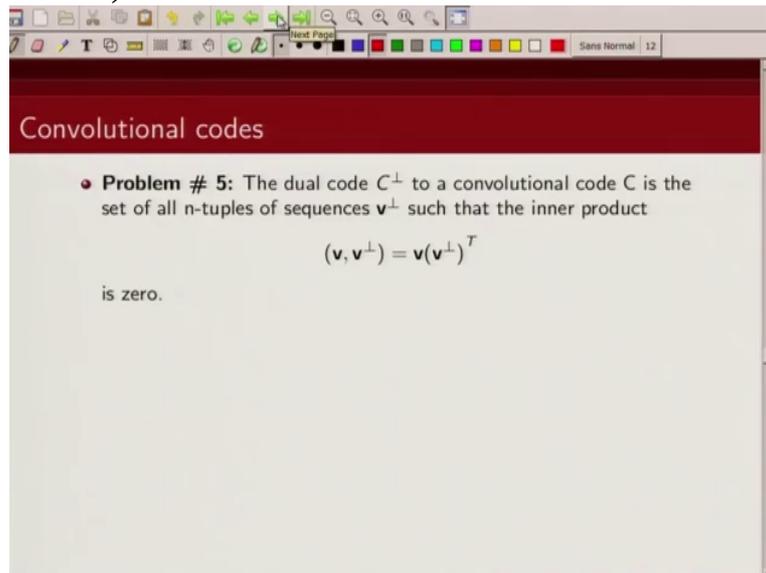
So let us say we have a generator matrix which is given by $G(D)$ and this be any realizable and delay free generator for this convolutional code C and let q be the least common multiple of all the denominators of the non zero entries of $G(D)$. So $G(D)$ basically has rational form and $q(D)$ is the l c m of the denominator terms of the non zero entries of $G(D)$. Now if we have such

(Refer Slide Time 13:02)

The screenshot shows a presentation slide with a dark red header containing the text "Convolutional codes". Below the header, there is a list of three bullet points. The first two are identical to the previous slide. The third bullet point is "Since $q(D)$ is a delayfree polynomial, we have". Below this, the equation $G'(D) = q(D)G(D)$ is displayed. Below the equation, it says "is a polynomial delayfree generator matrix for C."

an, and we will have a $q(D)$ which is basically delay free. So if we multiply our original generator matrix by this $q(D)$ we will end up with a new generator matrix $G'(D)$ which will be polynomial and delay free.

(Refer Slide Time 13:26)



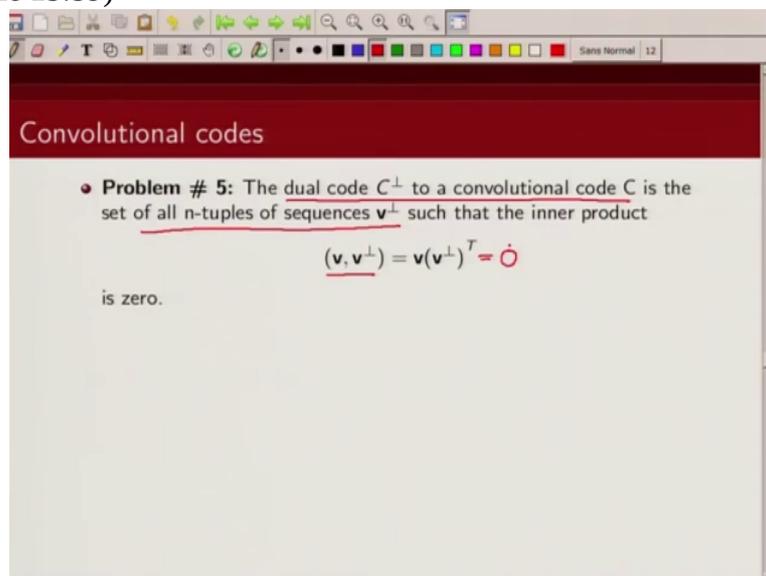
The screenshot shows a presentation slide with a red header containing the text "Convolutional codes". Below the header, there is a bullet point: "• Problem # 5: The dual code C^\perp to a convolutional code C is the set of all n-tuples of sequences v^\perp such that the inner product

$$(v, v^\perp) = v(v^\perp)^T$$

is zero.

Ok, so let's first define a dual of a convolutional code. So we define a dual of a convolutional, convolutional code c as a set of all end tuples of sequence v dual such that the inner product between these which is defined by v and v dual transpose, this is essentially zero. So

(Refer Slide Time 13:59)



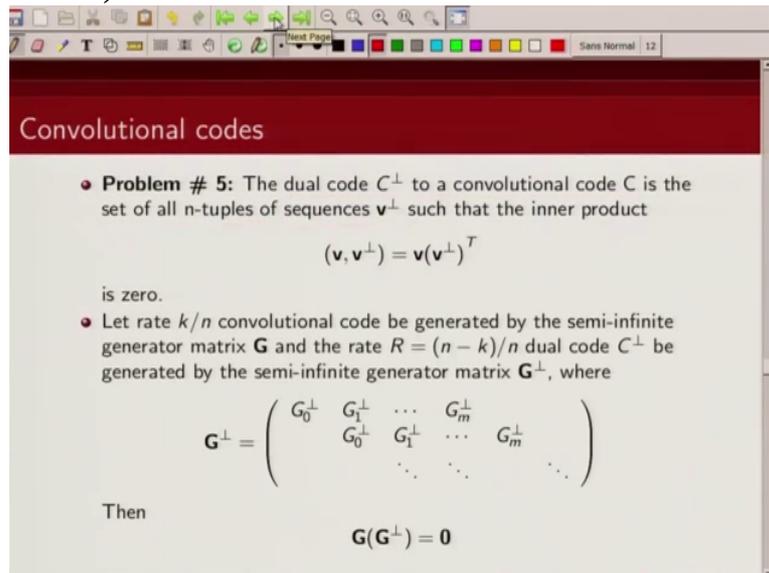
The screenshot shows a presentation slide with a red header containing the text "Convolutional codes". Below the header, there is a bullet point: "• Problem # 5: The dual code C^\perp to a convolutional code C is the set of all n-tuples of sequences v^\perp such that the inner product

$$(v, v^\perp) = v(v^\perp)^T = 0$$

is zero.

if you have a dual to the original convolutional code then if you take set of code n tuple sequence from the original code and the dual code their inner product will be zero.

(Refer Slide Time 14:14)



The slide is titled "Convolutional codes" and contains the following text:

- **Problem # 5:** The dual code C^\perp to a convolutional code C is the set of all n -tuples of sequences \mathbf{v}^\perp such that the inner product $(\mathbf{v}, \mathbf{v}^\perp) = \mathbf{v}(\mathbf{v}^\perp)^T$ is zero.
- Let rate k/n convolutional code be generated by the semi-infinite generator matrix \mathbf{G} and the rate $R = (n - k)/n$ dual code C^\perp be generated by the semi-infinite generator matrix \mathbf{G}^\perp , where

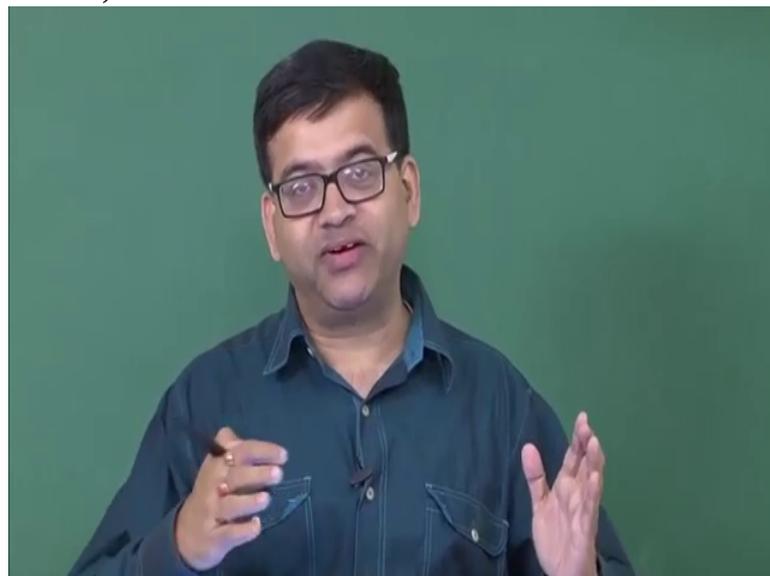
$$\mathbf{G}^\perp = \begin{pmatrix} G_0^\perp & G_1^\perp & \dots & G_m^\perp & & \\ & G_0^\perp & G_1^\perp & \dots & G_m^\perp & \\ & & \ddots & \ddots & & \ddots \end{pmatrix}$$

Then

$$\mathbf{G}(\mathbf{G}^\perp) = \mathbf{0}$$

So let us define a rate k by n convolutional code which is generated by a generator matrix \mathbf{G} and you know that we can write a generator

(Refer Slide Time 14:26)



matrix of convolutional code in a semi infinite fashion. And let's say

(Refer Slide Time 14:33)

Convolutional codes

- **Problem # 5:** The dual code C^\perp to a convolutional code C is the set of all n -tuples of sequences \mathbf{v}^\perp such that the inner product

$$(\mathbf{v}, \mathbf{v}^\perp) = \mathbf{v}(\mathbf{v}^\perp)^T$$
 is zero.
- Let rate k/n convolutional code be generated by the semi-infinite generator matrix \mathbf{G} and the rate $R = (n - k)/n$ dual code C^\perp be generated by the semi-infinite generator matrix \mathbf{G}^\perp , where

$$\mathbf{G}^\perp = \begin{pmatrix} G_0^\perp & G_1^\perp & \dots & G_m^\perp & & \\ & G_0^\perp & G_1^\perp & \dots & G_m^\perp & \\ & & \ddots & \ddots & & \ddots \end{pmatrix}$$

Then

$$\mathbf{G}(\mathbf{G}^\perp) = \mathbf{0}$$

the dual code is this which has rate n minus k by n and it is generated by this semi infinite generator matrix then show that $\mathbf{G} \mathbf{G}^\perp$ basically will be zero where as \mathbf{G} is the generator matrix of the original code and this is the generator matrix of the dual code.

(Refer Slide Time 15:05)

Convolutional codes

- Let $\mathbf{v} = \mathbf{u}\mathbf{G}$ and $\mathbf{v}^\perp = \mathbf{u}^\perp\mathbf{G}^\perp$, where \mathbf{v} and \mathbf{v}^\perp are orthogonal. Then we have

$$\mathbf{v}(\mathbf{v}^\perp)^T = \mathbf{u}\mathbf{G}(\mathbf{u}^\perp\mathbf{G}^\perp)^T = \mathbf{u}\mathbf{G}(\mathbf{G}^\perp)^T(\mathbf{u}^\perp)^T = \mathbf{0}$$

So how does this follows? So since $\mathbf{v} \mathbf{v}^\perp$ transpose is 0, so what is \mathbf{v} ? \mathbf{v} is \mathbf{u} times \mathbf{G} . And \mathbf{v}^\perp and \mathbf{v} dual is \mathbf{u}^\perp dual \mathbf{G}^\perp dual. Now we know that they will be dual of each other if $\mathbf{v} \mathbf{v}^\perp$ dual basically is orthogonal. So if this inner product is 0, now if inner product is zero, so then $\mathbf{u} \mathbf{G}$ and this transpose should be zero. So this we can write as $\mathbf{u} \mathbf{G} \mathbf{G}^\perp$ transpose. Now this term will be zero only if this term is zero

(Refer Slide Time 15:53)

Convolutional codes

- Let $\mathbf{v} = \mathbf{uG}$ and $\mathbf{v}^\perp = \mathbf{u}^\perp \mathbf{G}^\perp$, where \mathbf{v} and \mathbf{v}^\perp are orthogonal. Then we have

$$\mathbf{v}(\mathbf{v}^\perp)^T = \mathbf{uG}(\mathbf{u}^\perp \mathbf{G}^\perp)^T = \mathbf{uG}(\mathbf{G}^\perp)^T(\mathbf{u}^\perp)^T = \mathbf{0}$$

for all \mathbf{u} , Ok and hence

(Refer Slide Time 16:00)

Convolutional codes

- Let $\mathbf{v} = \mathbf{uG}$ and $\mathbf{v}^\perp = \mathbf{u}^\perp \mathbf{G}^\perp$, where \mathbf{v} and \mathbf{v}^\perp are orthogonal. Then we have

$$\mathbf{v}(\mathbf{v}^\perp)^T = \mathbf{uG}(\mathbf{u}^\perp \mathbf{G}^\perp)^T = \mathbf{uG}(\mathbf{G}^\perp)^T(\mathbf{u}^\perp)^T = \mathbf{0}$$

- Thus we have

$$\mathbf{G}(\mathbf{G}^\perp)^T = \mathbf{0}$$

we get this condition that this should be zero, Ok.

(Refer Slide Time 16:11)

Convolutional codes

- Let $\mathbf{v} = \mathbf{u}\mathbf{G}$ and $\mathbf{v}^\perp = \mathbf{u}^\perp\mathbf{G}^\perp$, where \mathbf{v} and \mathbf{v}^\perp are orthogonal. Then we have

$$\mathbf{v}(\mathbf{v}^\perp)^\mathbf{T} = \mathbf{u}\mathbf{G}(\mathbf{u}^\perp\mathbf{G}^\perp)^\mathbf{T} = \mathbf{u}\mathbf{G}(\mathbf{G}^\perp)^\mathbf{T}(\mathbf{u}^\perp)^\mathbf{T} = \mathbf{0}$$
- Thus we have

$$\mathbf{G}(\mathbf{G}^\perp) = \mathbf{0}$$
- The convolutional dual code C^\perp to a convolutional code C which is encoded by the rate $R = k/n$ generator matrix $G(D)$ is the set of all codewords encoded by any rate $R = (n - k)/n$ generator matrix $G_\perp(D)$ such that

$$\mathbf{G}(D)\mathbf{G}_\perp^\mathbf{T}(D) = \mathbf{0}$$

So a convolutional dual to a convolutional code c which is encoded by a rate k by n generator matrix $G(D)$ is set of all codewords encoded by rate $n - k$ with generator matrix $G_\perp(D)$ such that $G(D)$ and the generator matrix of the dual should be 0. So the dual of the code is defined by this.

Next,

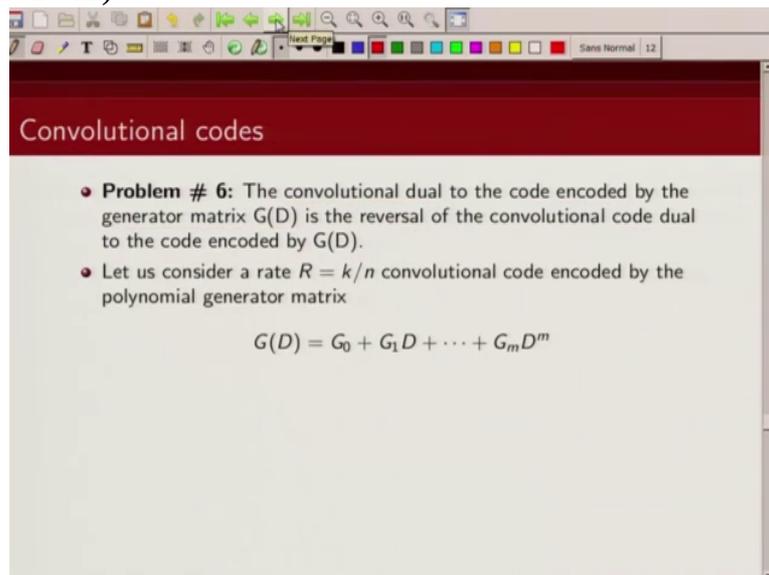
(Refer Slide Time 16:47)

Convolutional codes

- Problem # 6:** The convolutional dual to the code encoded by the generator matrix $G(D)$ is the reversal of the convolutional code dual to the code encoded by $G(D)$.

so here we will show that the convolutional dual to the code encoded by the generator matrix $G(D)$ is nothing but reversal of the convolutional code dual to the code generated by $G(D)$.
So

(Refer Slide Time 17:11)



The screenshot shows a presentation slide with a red header containing the text "Convolutional codes". Below the header, there are two bullet points:

- **Problem # 6:** The convolutional dual to the code encoded by the generator matrix $G(D)$ is the reversal of the convolutional code dual to the code encoded by $G(D)$.
- Let us consider a rate $R = k/n$ convolutional code encoded by the polynomial generator matrix

$$G(D) = G_0 + G_1D + \dots + G_mD^m$$

let us consider a rate k by n convolutional code which is encoded by a polynomial generator. We have already shown that we can find an equivalent polynomial delay free representation of any convolutional code, right. So let's see this is our

(Refer Slide Time 17:35)



$G(D)$ for original

(Refer Slide Time 17:37)

The slide is titled "Convolutional codes" and contains the following text:

- **Problem # 6:** The convolutional dual to the code encoded by the generator matrix $G(D)$ is the reversal of the convolutional code dual to the code encoded by $G(D)$.
- Let us consider a rate $R = k/n$ convolutional code encoded by the polynomial generator matrix

$$G(D) = G_0 + G_1D + \dots + G_mD^m$$

code c.

(Refer Slide Time 17:40)

The slide is titled "Convolutional codes" and contains the following text:

- **Problem # 6:** The convolutional dual to the code encoded by the generator matrix $G(D)$ is the reversal of the convolutional code dual to the code encoded by $G(D)$.
- Let us consider a rate $R = k/n$ convolutional code encoded by the polynomial generator matrix

$$G(D) = G_0 + G_1D + \dots + G_mD^m$$

- Let $\tilde{G}^\perp(D)$ denote the rate $R=(n - k)/n$ polynomial generator matrix

$$\tilde{G}^\perp(D) = G_m^\perp + G_{m-1}^\perp D + \dots + G_0^\perp D^{m^\perp}$$

which is the reciprocal of the generator matrix

$$G^\perp(D) = G_0^\perp + G_1^\perp D + \dots + G_m^\perp D^m$$

for the dual code C^\perp .

Now we define this as the reciprocal of the generator matrix. Now how do we come up with a reciprocal? So we do $G D$ minus 1 and we multiply by the maximum degree which is, in this case, m . This is how we get the reciprocal. This is how we get the reciprocal, Ok. Now

(Refer Slide Time 18:15)

Convolutional codes

- **Problem # 6:** The convolutional dual to the code encoded by the generator matrix $G(D)$ is the reversal of the convolutional code dual to the code encoded by $G(D)$.
- Let us consider a rate $R = k/n$ convolutional code encoded by the polynomial generator matrix $\vec{G}(D) = D^m G(D^{-1})$

$$G(D) = G_0 + G_1 D + \dots + G_m D^m$$
- Let $\vec{G}^\perp(D)$ denote the rate $R = (n - k)/n$ polynomial generator matrix

$$\vec{G}^\perp(D) = G_m^\perp + G_{m-1}^\perp D + \dots + G_0^\perp D^{m-1}$$
 which is the reciprocal of the generator matrix

$$G^\perp(D) = G_0^\perp + G_1^\perp D + \dots + G_m^\perp D^m$$
 for the dual code C^\perp .

what I have shown you here is the reciprocal of

(Refer Slide Time 18:22)

Convolutional codes

- **Problem # 6:** The convolutional dual to the code encoded by the generator matrix $G(D)$ is the reversal of the convolutional code dual to the code encoded by $G(D)$.
- Let us consider a rate $R = k/n$ convolutional code encoded by the polynomial generator matrix $\vec{G}(D) = D^m G(D^{-1})$

$$G(D) = G_0 + G_1 D + \dots + G_m D^m$$
- Let $\vec{G}^\perp(D)$ denote the rate $R = (n - k)/n$ polynomial generator matrix

$$\vec{G}^\perp(D) = G_m^\perp + G_{m-1}^\perp D + \dots + G_0^\perp D^{m-1}$$
 which is the reciprocal of the generator matrix

$$G^\perp(D) = G_0^\perp + G_1^\perp D + \dots + G_m^\perp D^m$$
 for the dual code C^\perp .

the generator matrix for the dual.

(Refer Slide Time 18:28)

Convolutional codes

- **Problem # 6:** The convolutional dual to the code encoded by the generator matrix $G(D)$ is the reversal of the convolutional code dual to the code encoded by $G(D)$.
- Let us consider a rate $R = k/n$ convolutional code encoded by the polynomial generator matrix $\vec{G}(D) = D^m G(D^{-1})$

$$G(D) = G_0 + G_1 D + \dots + G_m D^m$$

- Let $\vec{G}^\perp(D)$ denote the rate $R = (n - k)/n$ polynomial generator matrix

$$\vec{G}^\perp(D) = G_m^\perp + G_{m-1}^\perp D + \dots + G_0^\perp D^{m^\perp}$$

which is the reciprocal of the generator matrix

$$G^\perp(D) = G_0^\perp + G_1^\perp D + \dots + G_{m^\perp}^\perp$$

for the dual code C^\perp .

This is the generator matrix of the dual code. And this is the reciprocal of the generator matrix of the dual code. Now note we have to show that the convolutional dual to the code encoded by $G D$ is nothing but the reversal of the convolutional dual coded by $G D$. So if that's the case then $G D$, dot product of $G D$ with this should be zero. So

(Refer Slide Time 19:08)

Convolutional codes

- Then we have

$$\begin{aligned} G(D)(\vec{G}^\perp(D))^T &= G_0(G_{m^\perp}^\perp)^T + G_0(G_{m^\perp-1}^\perp)^T + G_1(G_{m^\perp}^\perp)^T D \\ &\quad + \dots + G_m(G_0^\perp)^T D^{m+m^\perp} \\ &= \left(\sum_{j=-m}^{m^\perp} \left(\sum_{i=0}^m G_i(G_{i+j}^\perp)^T \right) \right) D^{m+j} = 0 \end{aligned}$$

let's try to find $G D$ and transpose of the reversal of the

(Refer Slide Time 19:20)

Convolutional codes

- **Problem # 6:** The convolutional dual to the code encoded by the generator matrix $G(D)$ is the reversal of the convolutional code dual to the code encoded by $G(D)$.
- Let us consider a rate $R = k/n$ convolutional code encoded by the polynomial generator matrix $\vec{G}(D) = D^m G(D^{-1})$

$$G(D) = G_0 + G_1 D + \dots + G_m D^m$$
- Let $\vec{G}^\perp(D)$ denote the rate $R=(n-k)/n$ polynomial generator matrix

$$\vec{G}^\perp(D) = G_m^\perp + G_{m-1}^\perp D + \dots + G_0^\perp D^{m^\perp}$$
 which is the reciprocal of the generator matrix

$$G^\perp(D) = G_0^\perp + G_1^\perp D + \dots + G_{m^\perp}^\perp$$
 for the dual code C^\perp .

dual matrix. See; again pay attention to the question. Basically what we are saying is the dual of the code encoded by the generator matrix is given by the reversal of the convolutional code dual. So reversal of the convolutional code dual is this, right. So if the dual of the convolutional code is given by reversal of the convolutional code, then what is the property, the, this generator matrix should satisfy? This and this transpose should be zero. So we have to show that $G D$ and this generator matrix transpose is zero.

(Refer Slide Time 20:07)

Convolutional codes

- Then we have

$$\begin{aligned} \underline{G(D)(\vec{G}^\perp(D))^\top} &= G_0(G_{m^\perp}^\perp)^\top + G_0(G_{m-1}^\perp)^\top + G_1(G_{m^\perp}^\perp)^\top \\ &\quad + \dots + G_m(G_0^\perp)^\top D^{m+m^\perp} \\ &= \left(\sum_{j=-m}^{m^\perp} \left(\sum_{i=0}^m G_i(G_{i+j}^\perp)^\top \right) \right) D^{m+j} = 0 \end{aligned}$$

So we do this. So this can be written as this term, Ok?

(Refer Slide Time 20:16)

Convolutional codes

- Then we have

$$\underline{G(D)(\tilde{G}(D))^T} = G_0(G_{m+1}^\perp)^T + G_0(G_{m+1}^\perp)^T + G_1(G_{m+1}^\perp)D + \dots + G_m(G_0^\perp)^T D^{m+m+1}$$

$$= \left(\sum_{j=-m}^{m+1} \left(\sum_{i=0}^m G_i(G_{i+j}^\perp)^T \right) \right) D^{m+j} = 0$$

Further we can write this as double summation

(Refer Slide Time 20:24)

Convolutional codes

- Then we have

$$\underline{G(D)(\tilde{G}(D))^T} = G_0(G_{m+1}^\perp)^T + G_0(G_{m+1}^\perp)^T + G_1(G_{m+1}^\perp)D + \dots + G_m(G_0^\perp)^T D^{m+m+1}$$

$$= \left(\sum_{j=-m}^{m+1} \left(\sum_{i=0}^m G_i(G_{i+j}^\perp)^T \right) \right) D^{m+j} = 0$$

and what is this? This is the dual of, generator matrix is dual of this, so this, this G D G transpose should be zero, right? So this whole summation would be also zero. So what we have shown is then the dual of the, dual of this code

(Refer Slide Time 20:50)

Convolutional codes

- Then we have

$$G(D)(\tilde{G}(D))^T = G_0(G_{m-1}^\perp)^T + G_0(G_{m-2}^\perp)^T + G_1(G_{m-1}^\perp)D + \dots + G_m(G_0^\perp)^T D^{m+m^\perp}$$

$$= \left(\sum_{j=-m}^{m^\perp} \left(\sum_{i=0}^m G_i(G_{i+j}^\perp)^T \right) \right) D^{m+j} = 0$$

which has generator matrix G

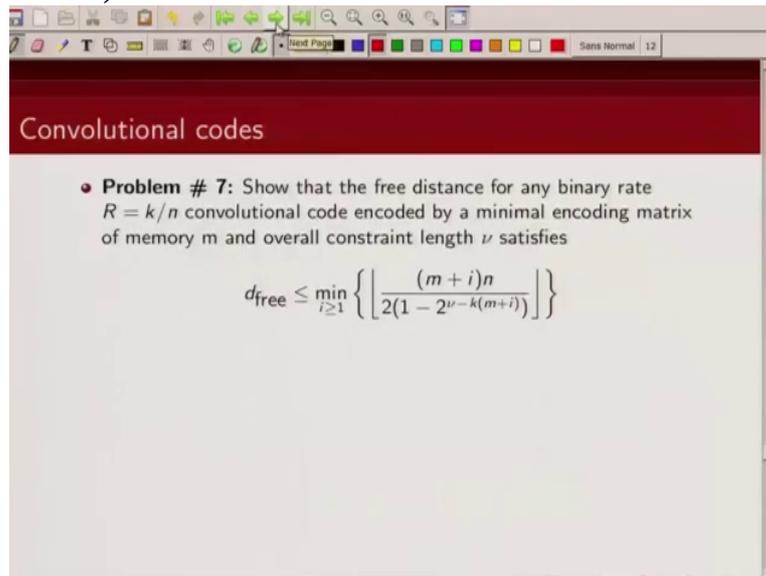
(Refer Slide Time 20:53)

Convolutional codes

- Problem # 6:** The convolutional dual to the code encoded by the generator matrix $G(D)$ is the reversal of the convolutional code dual to the code encoded by $G(D)$.
- Let us consider a rate $R = k/n$ convolutional code encoded by the polynomial generator matrix $G(D) = G_0 + G_1 D + \dots + G_m D^m$
- Let $\tilde{G}^\perp(D)$ denote the rate $R = (n-k)/n$ polynomial generator matrix $\tilde{G}^\perp(D) = G_m^\perp + G_{m-1}^\perp D + \dots + G_0 D^{m^\perp}$ which is the reciprocal of the generator matrix $G^\perp(D) = G_0^\perp + G_1^\perp D + \dots + G_m^\perp D^m$ for the dual code C^\perp .

is nothing but the reversal of convolutional code dual because this generator matrix is nothing but it is the reversal of the generator matrix of the dual code of c, Ok.

(Refer Slide Time 21:14)



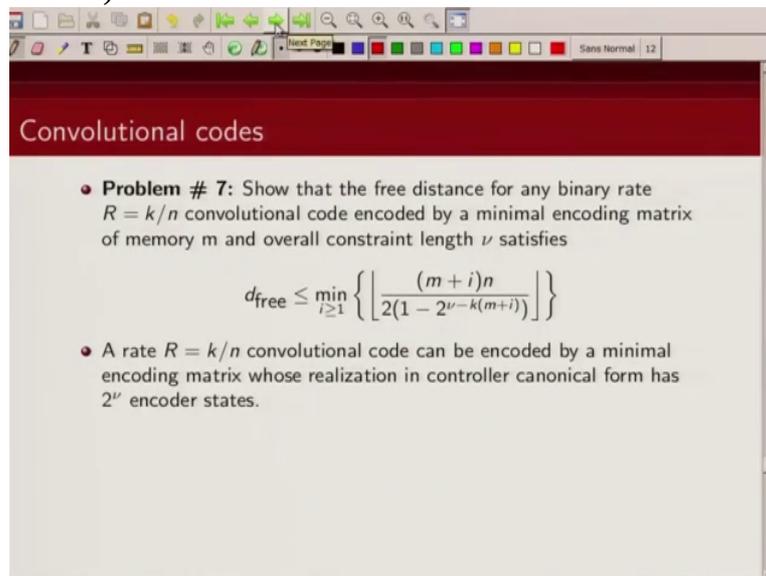
The slide is titled "Convolutional codes" and contains the following text:

- **Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \left\lfloor \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rfloor \right\}$$

Next we are going to show a bound on free distance of convolutional code. So show that if you have a rate k by n convolutional code whose encoding matrix is memory n and overall constraint length μ then free distance is upper bounded by this.

(Refer Slide Time 21:43)



The slide is titled "Convolutional codes" and contains the following text:

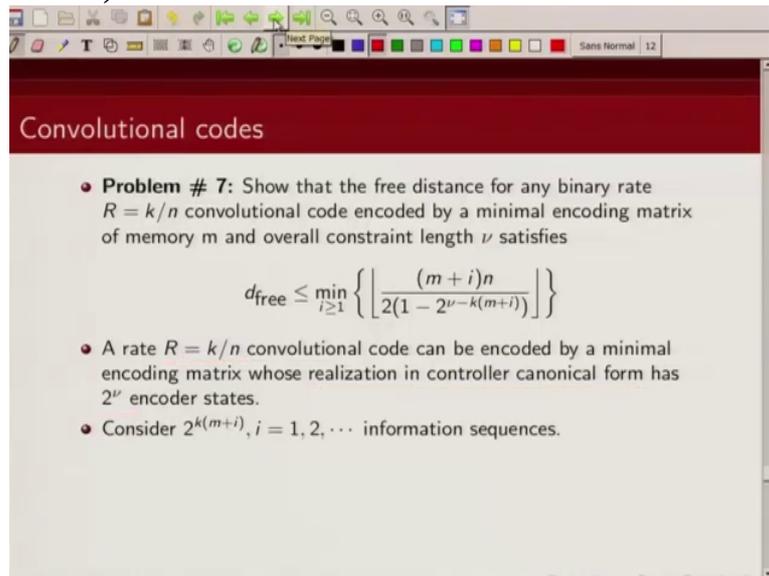
- **Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \left\lfloor \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rfloor \right\}$$

- A rate $R = k/n$ convolutional code can be encoded by a minimal encoding matrix whose realization in controller canonical form has 2^ν encoder states.

So if you have a rate k by n code, we can realize it using controller canonical form using 2^ν encoder state because overall constraint length is 2^ν .

(Refer Slide Time 22:00)



The slide is titled "Convolutional codes" and contains the following content:

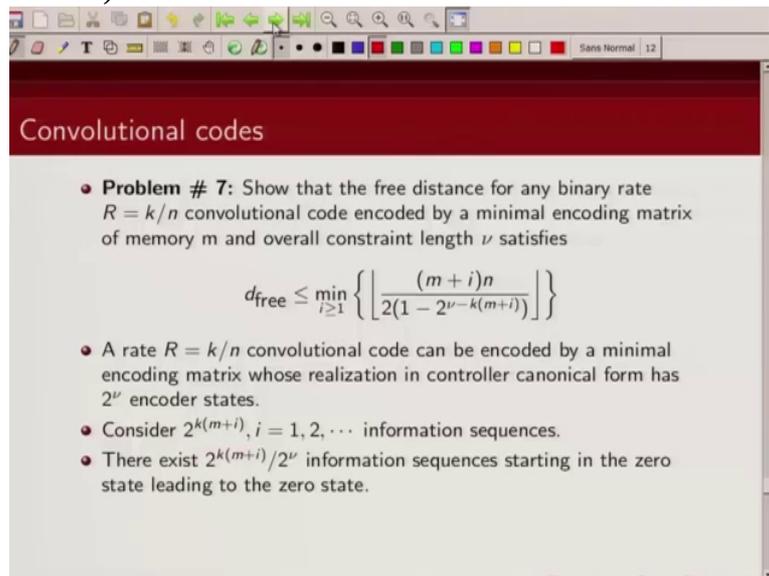
- **Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \left\lfloor \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rfloor \right\}$$

- A rate $R = k/n$ convolutional code can be encoded by a minimal encoding matrix whose realization in controller canonical form has 2^ν encoder states.
- Consider $2^{k(m+i)}$, $i = 1, 2, \dots$ information sequences.

Now if we consider 2^k raised to power m plus i information sequences

(Refer Slide Time 22:08)



The slide is titled "Convolutional codes" and contains the following content:

- **Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \left\lfloor \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rfloor \right\}$$

- A rate $R = k/n$ convolutional code can be encoded by a minimal encoding matrix whose realization in controller canonical form has 2^ν encoder states.
- Consider $2^{k(m+i)}$, $i = 1, 2, \dots$ information sequences.
- There exist $2^{k(m+i)}/2^\nu$ information sequences starting in the zero state leading to the zero state.

then there exist $2^k m$ plus i divided by 2^ν information sequence starting at zero state leading to all zero state. So

(Refer Slide Time 22:20)

Convolutional codes

- **Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\lfloor \left\lceil \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rceil \right\rfloor$$
- A rate $R = k/n$ convolutional code can be encoded by a minimal encoding matrix whose realization in controller canonical form has 2^ν encoder states.
- Consider $2^{k(m+i)}$, $i = 1, 2, \dots$ information sequences.
- There exist $2^{k(m+i)}/2^\nu$ information sequences starting in the zero state leading to the zero state.
- Corresponding code sequences constitute a block code with $M = 2^{k(m+i)-\nu}$ codewords and blocklength $N = (m+i)n$ for $i = 1, 2, \dots$.

if we count number of such codewords, then number of such codewords m is given by 2 raised to power k into m plus i minus ν and what is the length of this codeword? Length of this codeword would be length plus i into n , because it is a rate k by n code, right?

Next we are going to use

(Refer Slide Time 22:49)

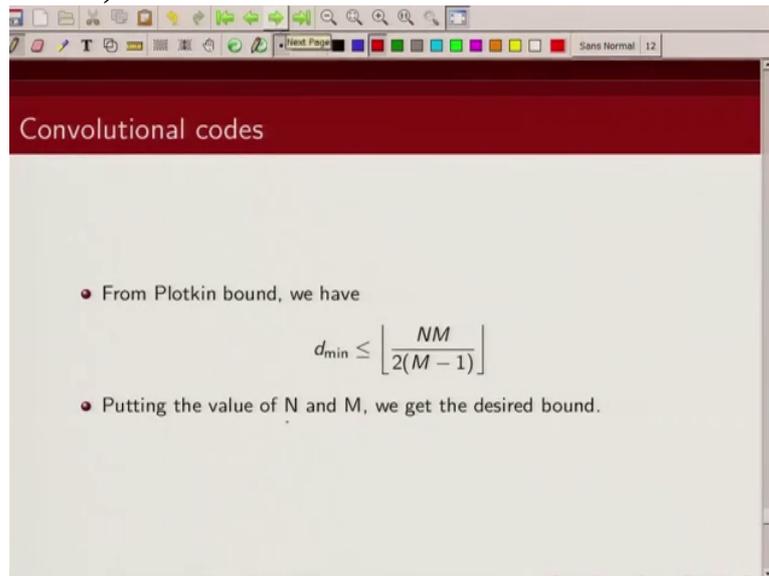
Convolutional codes

- From Plotkin bound, we have

$$d_{\text{min}} \leq \left\lfloor \frac{NM}{2(M-1)} \right\rfloor$$

Plotkin's bound. The Plotkin bound says it upper bounds the minimum distance as floor of, this is codewords length, number of codewords, 2 into number of codewords minus 1 . So in this example what is our

(Refer Slide Time 23:07)



Convolutional codes

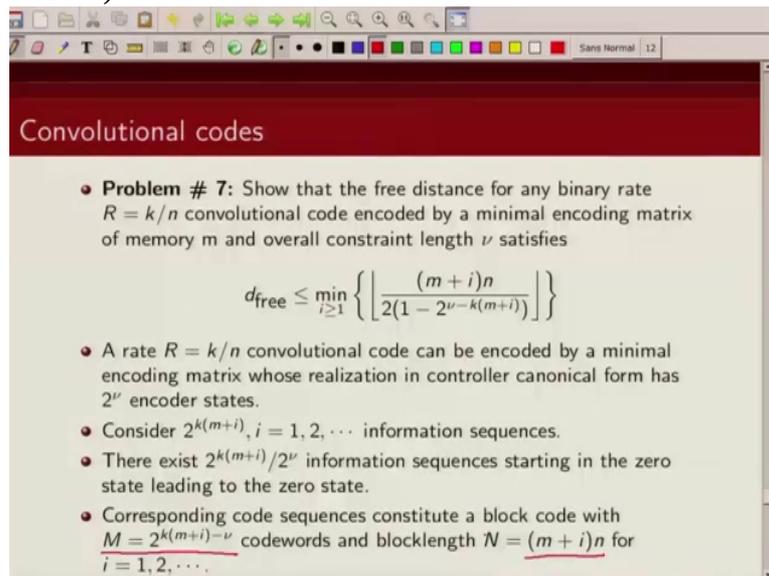
- From Plotkin bound, we have

$$d_{\min} \leq \left\lfloor \frac{NM}{2(M-1)} \right\rfloor$$

- Putting the value of N and M, we get the desired bound.

n? Our n is

(Refer Slide Time 23:10)



Convolutional codes

- Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \left\lfloor \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rfloor \right\}$$

- A rate $R = k/n$ convolutional code can be encoded by a minimal encoding matrix whose realization in controller canonical form has 2^ν encoder states.
- Consider $2^{k(m+i)}$, $i = 1, 2, \dots$ information sequences.
- There exist $2^{k(m+i)}/2^\nu$ information sequences starting in the zero state leading to the zero state.
- Corresponding code sequences constitute a block code with $M = 2^{k(m+i)-\nu}$ codewords and blocklength $N = (m+i)n$ for $i = 1, 2, \dots$.

this, Ok?

(Refer Slide Time 23:14)

Convolutional codes

- **Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \left\lfloor \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rfloor \right\}$$

- A rate $R = k/n$ convolutional code can be encoded by a minimal encoding matrix whose realization in controller canonical form has 2^ν encoder states.
- Consider $2^{k(m+i)}$, $i = 1, 2, \dots$ information sequences.
- There exist $2^{k(m+i)}/2^\nu$ information sequences starting in the zero state leading to the zero state.
- Corresponding code sequences constitute a block code with $M = 2^{k(m+i)-\nu}$ codewords and blocklength $N = (m+i)n$ for $i = 1, 2, \dots$.

This is our n. And what is our m? Our m is this.

(Refer Slide Time 23:19)

Convolutional codes

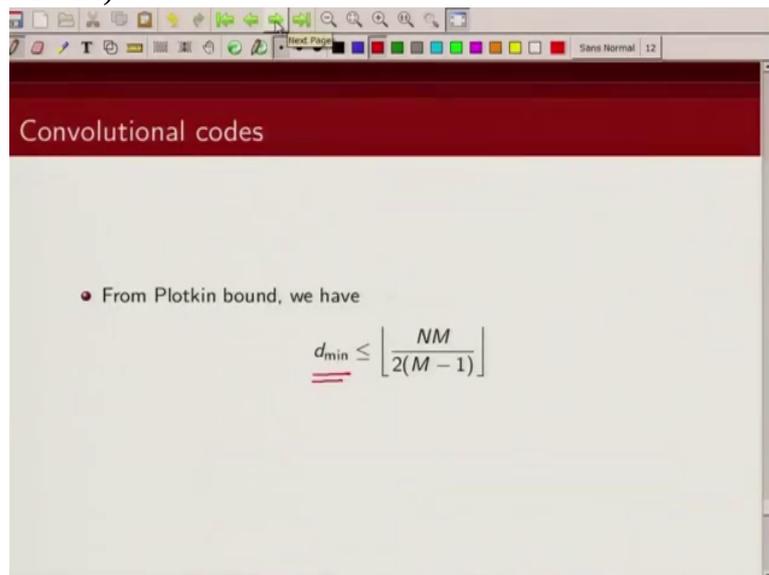
- **Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \left\lfloor \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rfloor \right\}$$

- A rate $R = k/n$ convolutional code can be encoded by a minimal encoding matrix whose realization in controller canonical form has 2^ν encoder states.
- Consider $2^{k(m+i)}$, $i = 1, 2, \dots$ information sequences.
- There exist $2^{k(m+i)}/2^\nu$ information sequences starting in the zero state leading to the zero state.
- Corresponding code sequences constitute a block code with $M = 2^{k(m+i)-\nu}$ codewords and blocklength $N = (m+i)n$ for $i = 1, 2, \dots$.

So we plug in this value of m and n in our

(Refer Slide Time 23:24)



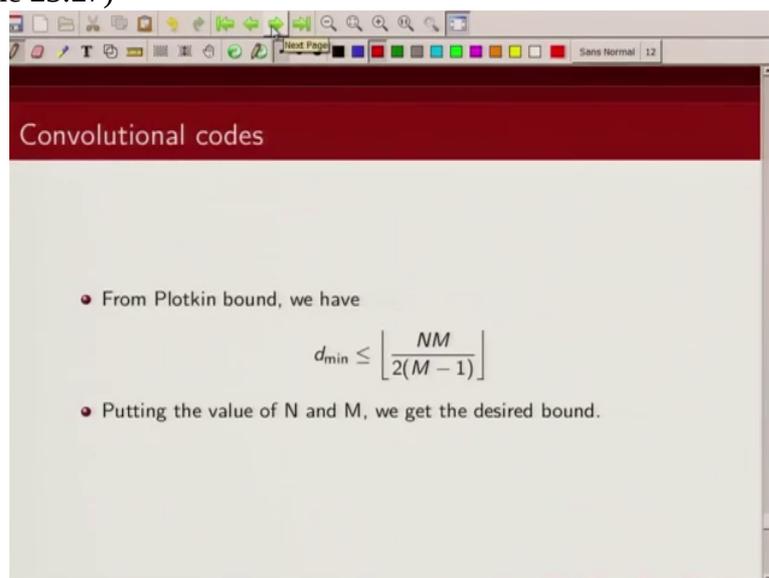
Convolutional codes

- From Plotkin bound, we have

$$d_{\min} \leq \left\lfloor \frac{NM}{2(M-1)} \right\rfloor$$

Plotkin bounds what we will get is

(Refer Slide Time 23:27)



Convolutional codes

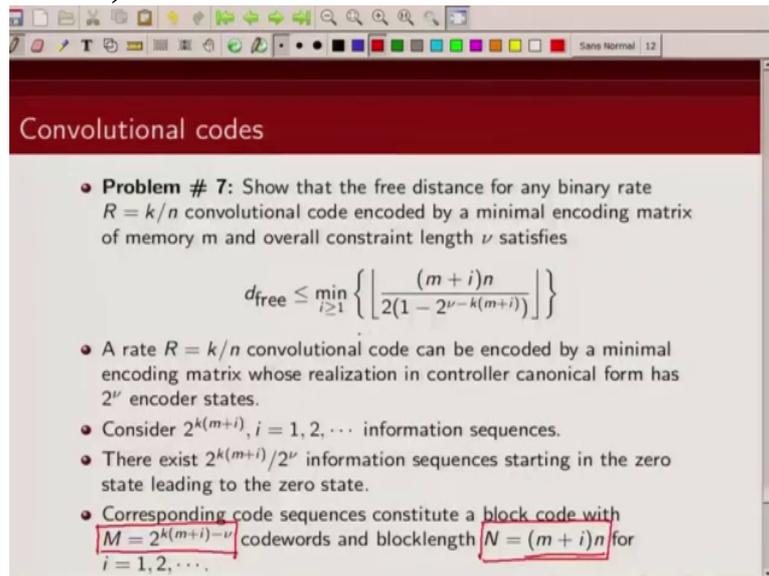
- From Plotkin bound, we have

$$d_{\min} \leq \left\lfloor \frac{NM}{2(M-1)} \right\rfloor$$

- Putting the value of N and M, we get the desired bound.

our desired bound which is this result

(Refer Slide Time 23:31)

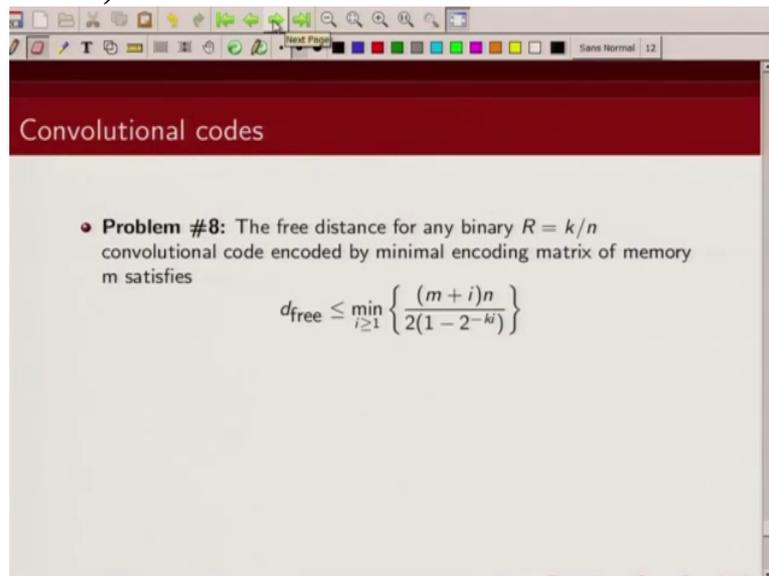


The slide is titled "Convolutional codes" and contains the following content:

- **Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies
$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \left\lfloor \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rfloor \right\}$$
- A rate $R = k/n$ convolutional code can be encoded by a minimal encoding matrix whose realization in controller canonical form has 2^ν encoder states.
- Consider $2^{k(m+i)}$, $i = 1, 2, \dots$ information sequences.
- There exist $2^{k(m+i)}/2^\nu$ information sequences starting in the zero state leading to the zero state.
- Corresponding code sequences constitute a block code with $M = 2^{k(m+i)-\nu}$ codewords and blocklength $N = (m+i)n$ for $i = 1, 2, \dots$.

, Ok

(Refer Slide Time 23:35)

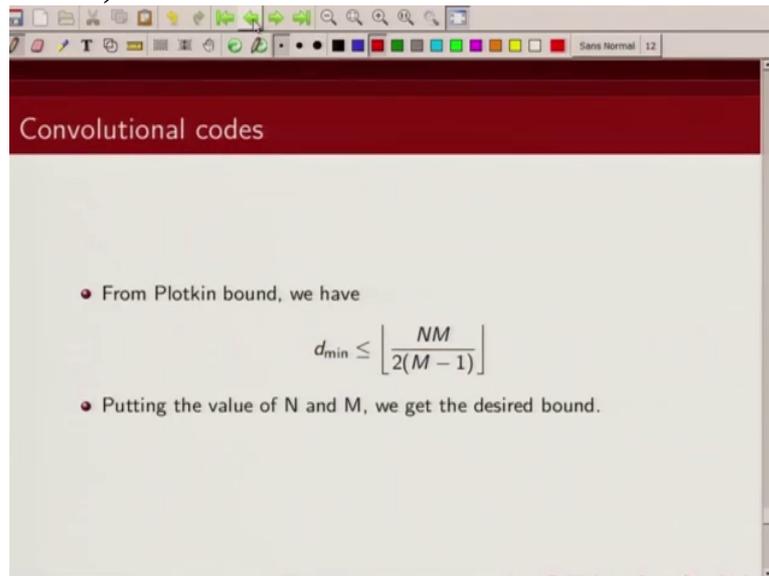


The slide is titled "Convolutional codes" and contains the following content:

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies
$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\}$$

So the next problem we will solve is to show that free distance of a rate k by n convolutional code is upper bounded by this. Now this result can be obtained from

(Refer Slide Time 23:54)



Convolutional codes

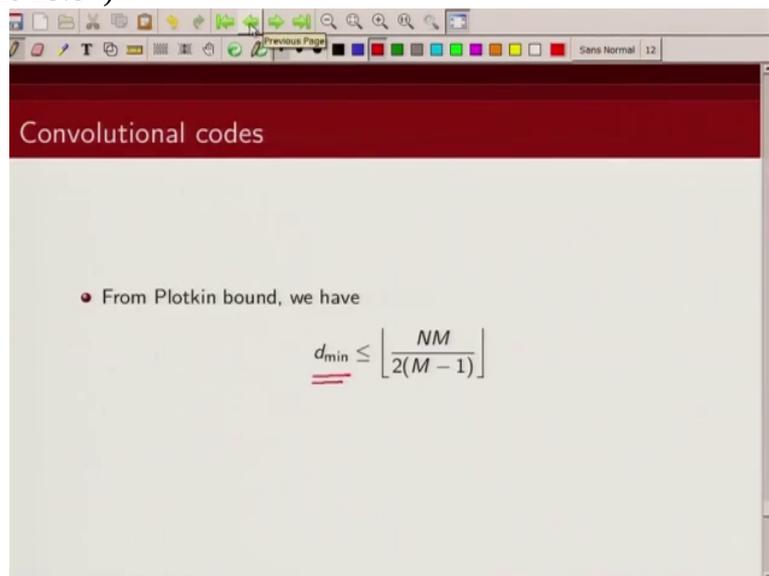
- From Plotkin bound, we have

$$d_{\min} \leq \left\lfloor \frac{NM}{2(M-1)} \right\rfloor$$

- Putting the value of N and M, we get the desired bound.

the bound that we had

(Refer Slide Time 23:57)



Convolutional codes

- From Plotkin bound, we have

$$\underline{\underline{d_{\min}}} \leq \left\lfloor \frac{NM}{2(M-1)} \right\rfloor$$

derived earlier, this bound,

(Refer Slide Time 24:02)

The slide is titled "Convolutional codes" and contains the following content:

- **Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies
$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \left\lfloor \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rfloor \right\}$$
- A rate $R = k/n$ convolutional code can be encoded by a minimal encoding matrix whose realization in controller canonical form has 2^ν encoder states.
- Consider $2^{k(m+i)}$, $i = 1, 2, \dots$ information sequences.
- There exist $2^{k(m+i)}/2^\nu$ information sequences starting in the zero state leading to the zero state.
- Corresponding code sequences constitute a block code with $M = 2^{k(m+i)-\nu}$ codewords and blocklength $N = (m+i)n$ for $i = 1, 2, \dots$.

Ok.

(Refer Slide Time 24:04)

The slide is titled "Convolutional codes" and contains the following content:

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies
$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\}$$
- Also
$$\lim_{m \rightarrow \infty} \frac{d_{\text{free}}}{mn} \leq \frac{1}{2}$$

And also we will show that this relation holds. So since

(Refer Slide Time 24:10)

Convolutional codes

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\}$$
- Also

$$\lim_{m \rightarrow \infty} \frac{d_{\text{free}}}{mn} \leq \frac{1}{2}$$
- **Solutions:** Since $\nu \leq km$, the bound follows from the result of the last question.

overall constraint length is less than k times m , then if you go back to this expression

(Refer Slide Time 24:19)

Convolutional codes

- **Problem # 7:** Show that the free distance for any binary rate $R = k/n$ convolutional code encoded by a minimal encoding matrix of memory m and overall constraint length ν satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \left\lfloor \frac{(m+i)n}{2(1-2^{\nu-k(m+i)})} \right\rfloor \right\}$$
- A rate $R = k/n$ convolutional code can be encoded by a minimal encoding matrix whose realization in controller canonical form has 2^ν encoder states.
- Consider $2^{k(m+i)}$, $i = 1, 2, \dots$ information sequences.
- There exist $2^{k(m+i)}/2^\nu$ information sequences starting in the zero state leading to the zero state.
- Corresponding code sequences constitute a block code with $M = 2^{k(m+i)-\nu}$ codewords and blocklength $N = (m+i)n$ for $i = 1, 2, \dots$.

ν minus k m is actually less than zero so we can then upper bound this by just 2 raised to power k i . If we do that we get this expression, Ok.

(Refer Slide Time 24:38)

The slide is titled "Convolutional codes" and contains the following text:

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\}$$

- Also

$$\lim_{m \rightarrow \infty} \frac{d_{\text{free}}}{mn} \leq \frac{1}{2}$$

(Refer Slide Time 24:39)

The slide is titled "Convolutional codes" and contains the following text:

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\}$$

- Also

$$\lim_{m \rightarrow \infty} \frac{d_{\text{free}}}{mn} \leq \frac{1}{2}$$

- **Solutions:** Since $\nu \leq km$, the bound follows from the result of the last question.

Now how do we get this expression?

(Refer Slide Time 24:46)

The slide is titled "Convolutional codes" and contains the following text:

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\}$$
- Also

$$\lim_{m \rightarrow \infty} \frac{d_{\text{free}}}{mn} \leq \frac{1}{2}$$
- **Solutions:** Since $\nu \leq km$, the bound follows from the result of the last question.

So we know that this term is less than 1, and so this we can just upper bound by $m + i$ by n by 2, and if I

(Refer Slide Time 25:00)

The slide is titled "Convolutional codes" and contains the following text with handwritten annotations:

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\} \leq \frac{(m+i)n}{2}$$
- Also

$$\lim_{m \rightarrow \infty} \frac{d_{\text{free}}}{mn} \leq \frac{1}{2}$$
- **Solutions:** Since $\nu \leq km$, the bound follows from the result of the last question.

take m and n out, this will be $1 + i$ times m by 2.

(Refer Slide Time 25:10)

Convolutional codes

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies

$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\} \leq \frac{(m+i)n}{2}$$
- Also

$$\lim_{m \rightarrow \infty} \frac{d_{\text{free}}}{mn} \leq \frac{1}{2}$$
- **Solutions:** Since $\nu \leq km$, the bound follows from the result of the last question.

$$\frac{d_{\text{free}}}{mn} \leq \frac{1 + \frac{i}{m}}{2}$$

So d_{free} by m of n will then be, so if I do further, d_{free} by m of n would be upper bounded by $1 + \frac{i}{m}$ by m by 2

(Refer Slide Time 25:26)

Convolutional codes

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies

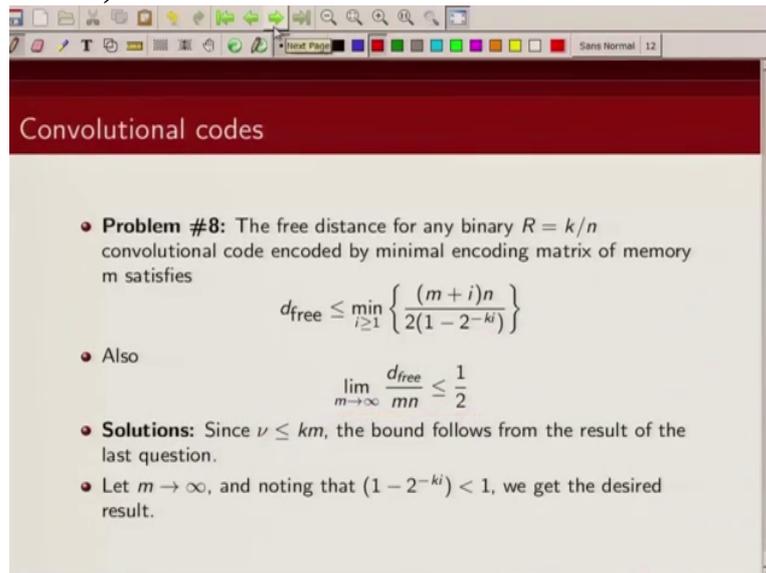
$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\} \leq \frac{(m+i)n}{2}$$
- Also

$$\lim_{m \rightarrow \infty} \frac{d_{\text{free}}}{mn} \leq \frac{1}{2}$$
- **Solutions:** Since $\nu \leq km$, the bound follows from the result of the last question.

$$\frac{d_{\text{free}}}{mn} \leq \frac{1 + \frac{i}{m}}{2}$$

and if we let m go to infinity this would go to zero so that would be upper bounded by half. So that's

(Refer Slide Time 25:34)

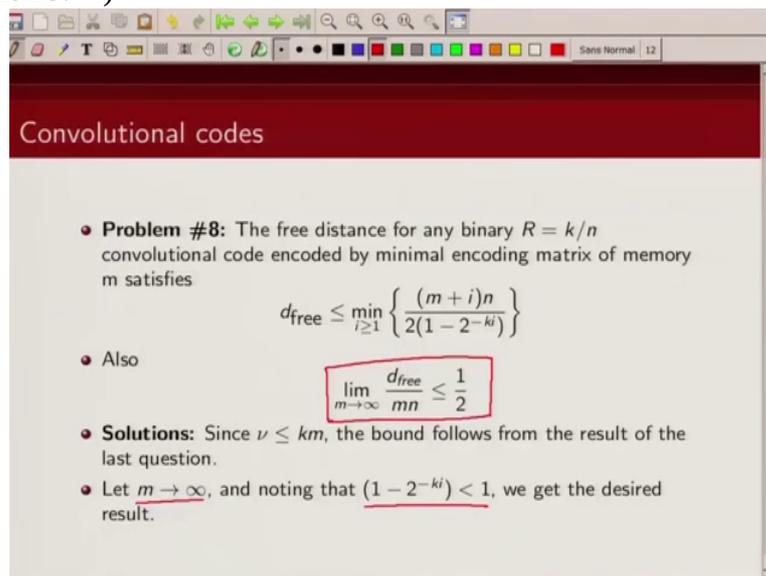


The slide is titled "Convolutional codes" and contains the following text:

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies
$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\}$$
- Also
$$\lim_{m \rightarrow \infty} \frac{d_{\text{free}}}{mn} \leq \frac{1}{2}$$
- **Solutions:** Since $\nu \leq km$, the bound follows from the result of the last question.
- Let $m \rightarrow \infty$, and noting that $(1 - 2^{-ki}) < 1$, we get the desired result.

the proof. So you let m go to infinity and since this is less than 1, what you will get is this,

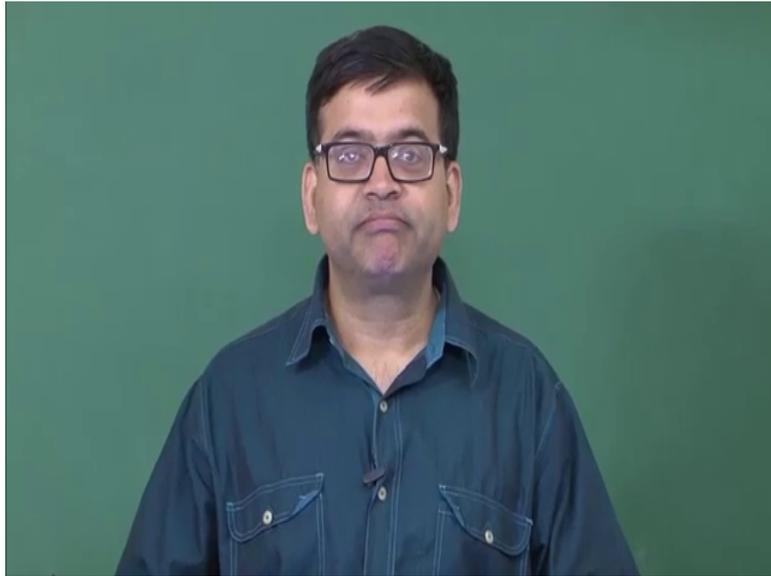
(Refer Slide Time 25:44)



The slide is titled "Convolutional codes" and contains the following text:

- **Problem #8:** The free distance for any binary $R = k/n$ convolutional code encoded by minimal encoding matrix of memory m satisfies
$$d_{\text{free}} \leq \min_{i \geq 1} \left\{ \frac{(m+i)n}{2(1-2^{-ki})} \right\}$$
- Also
$$\lim_{m \rightarrow \infty} \frac{d_{\text{free}}}{mn} \leq \frac{1}{2}$$
- **Solutions:** Since $\nu \leq km$, the bound follows from the result of the last question.
- Let $m \rightarrow \infty$, and noting that $(1 - 2^{-ki}) < 1$, we get the desired result.

(Refer Slide Time 25:45)



thank you