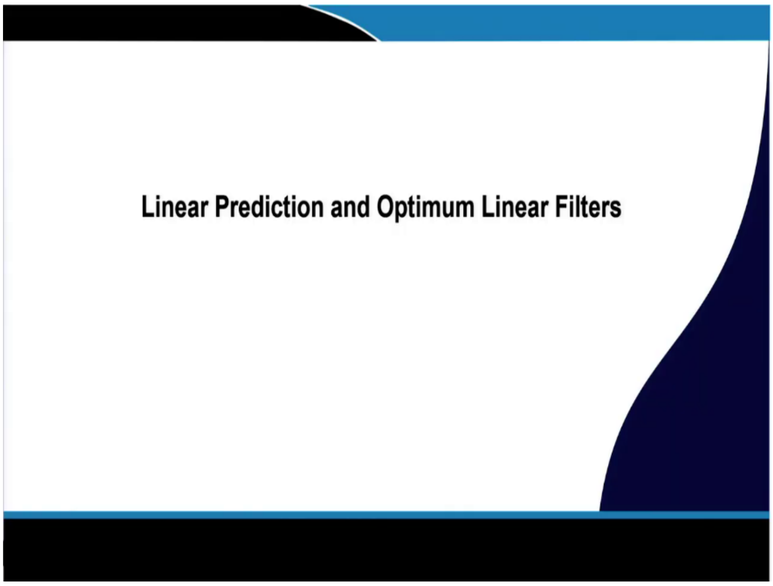
Signal Processing Techniques and Its Applications
Dr. Shyamal Kumar Das Mandal
Advanced Technology Development Centre
Indian Institute of Technology, Kharagpur

Lecture - 45
Linear Prediction and Optimum Linear Filters

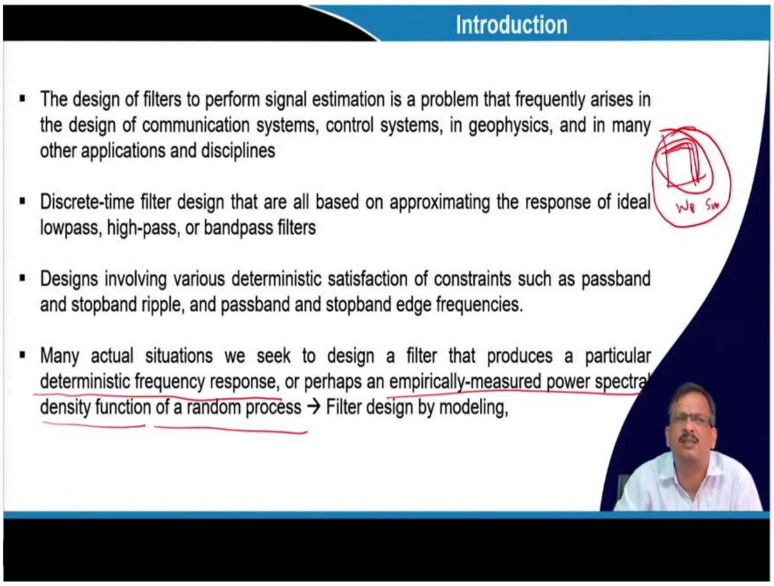
Ok. So, for the last two weeks, we have talked about the design of filters; we have talked about the FIR and IIR filters. So, a lot of things we have talked about.

(Refer Slide Time: 00:31)



So, this week, we will talk about linear prediction and optimal linear filtering. So, why did we design the filter? If you see that we have talk about the filter design, why we design the filter?

(Refer Slide Time: 00:47)



So, the design of the filter and filter design are performed for signal estimation. So, using the filter design, suppose I want to see whether, from 1 kilohertz or 2 kilohertz, the signal frequency component exists or not or estimate the signal within this frequency range. So, for the estimation of a signal, we definitely use the filter, which means many applications, such as communication systems, control systems, geophysics, and many other applications and disciplines. We use the filter to estimate the performance signal.

So, if you see when I say the low pass filter when I say band pass filter, this estimation of the signal, the design of that filter is based on the approximation of the ideal response of a low pass filter, high pass filter, band pass filter that is what. Now, suppose I said in a communication system that I want to estimate the signal and estimate the spectral content of the signal in between, let us say, 2 kilohertz to 4 kilohertz.

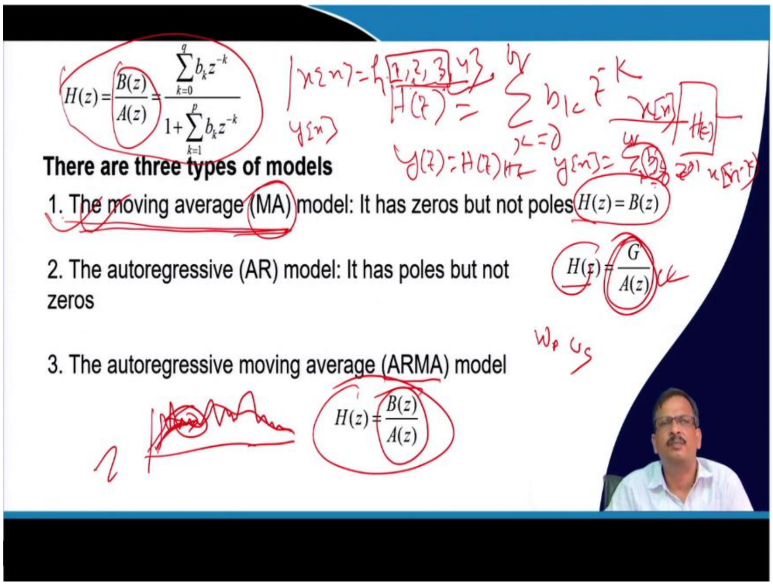
When I say that, that means I want to estimate the signal from 2 kilohertz to 4 kilohertz, ok. So, in that case, I have to design the filter. How do I design the filter? We consider a low-pass ideal filter and then try to approximate that approximation, which is the ideal frequency response of the desired filter. So, what are the design problems involved?

Now, this kind of design problem involves many constraints. What are the constraints? You know the pass band edge frequency, stopband edge frequency, passband ripple, stopband ripple, passband attenuation, and all kinds of design constraints are there. However, in many types and applications, I required a filter that produced a particular deterministic frequency response or perhaps empirically measured the power spectral density function of a random process.

So, I do not want that kind of constraint kind of thing. So, what do let us? I want to estimate the power spectral density of a particular frequency range. So, instead of designing an ideal filter using approximation methods, let us change that topic to a modelling technique.

So, modelling means that when you design and filter, what are we basically doing? From the desired frequency response, we try approximating the desired frequency response and find out the filter coefficient. Let us say that we do not want constrained methods; we want a modeling-based approach. Can I design a filter using modelling? So, what kind of modelling?

(Refer Slide Time: 03:51)



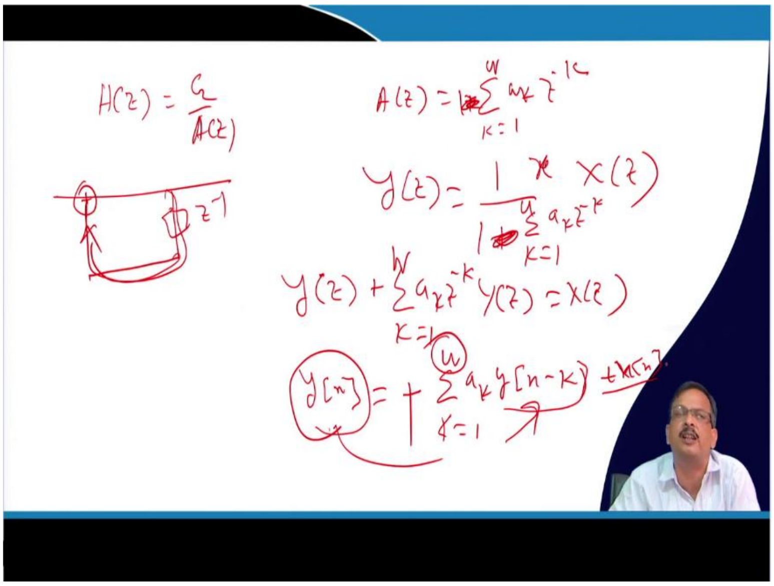
Now, what is a filter? If you see, what is a filter? Filter is nothing but a transfer function H(z), which is B(z) by A(z). So, B(z) is the pole position, B(z) is the zero position, and A(z) is the pole position. Now, let us say there are three types of models. What is the model? Moving average model MA model, MA. What is MA? MA is nothing but a when H(z) is equal to B(z) because if H(z) is equal to B(z), then I know H(z) is equal to the summation of k equal to 0 to q bk z^{-k}; only zeros are there, no poles.

So, when we multiply with output when I say I have a filter, which is H(z), which is H(z) is equal to B(z); if I apply a signal inside here, what I get outside is nothing but a multiplying signal. So, I can say y(z) is equal to H(z) multiplied by x(z), or I can say y[n] is equal to nothing but a summation of k equal to 0 to q bk into z^{-k} x[z], which is nothing but I can say z^{x} minus n minus k.

So, from the previous q number of samples, I am doing a moving average multiplied by a constraint bk, coefficient bk. So that is why it is called a moving average filter; that means suppose I have a signal x[n], so I can say that 1, 2, 3, 4, let us say this is my x[n], ok. So, I said that I have applied a filter, so I can say that y[n] is nothing but a moving average of, let us say, three samples. So, 1 plus 2 plus 3 with the multiplication of 1 will be multiplied by the b 1 or b 0, b 1 b 2 and then summed.

Then I can say 2 3 4 sum, then 3 4 let sum. So, it is a moving average kind of thing; that is why it is called the moving average model, then the autoregressive model, autoregression model. What is the autoregressive model? H(z) is equal to G by A(z).

(Refer Slide Time: 06:14)



So, I said H(z) is equal to G by A(z). So, my H(z) is equal to G by A(z); G is the gain,

$$A(z) = 1 + \sum_{k=1}^{q} a_k z^{-k}$$

So, when I apply a signal x[n], x[z], I can say y[z] is equal to nothing, but a 1 by 1 plus k equals 1 to q ak z^{-k} into x[z]. So, I can say it is nothing, but a y(z) plus k equal to 1 to q ak

z^{-k} into y(z) is equal to x(z). Now, when I say time domain signal, it is nothing, but a y[n] is equal to minus k equal to 1 to q ak y[n] minus k plus x[n].

So, when I want to implement it, I can say there is feedback, negative feedback, that y or n will be delayed by z^{-1} or minus 2 minus 3 minus k and fed to hear negative feedback if it is plus. If it is minus, let us say it is minus; then I can say it is nothing but a plus. So, I can say it is a feedback positive; let us give positive feedback.

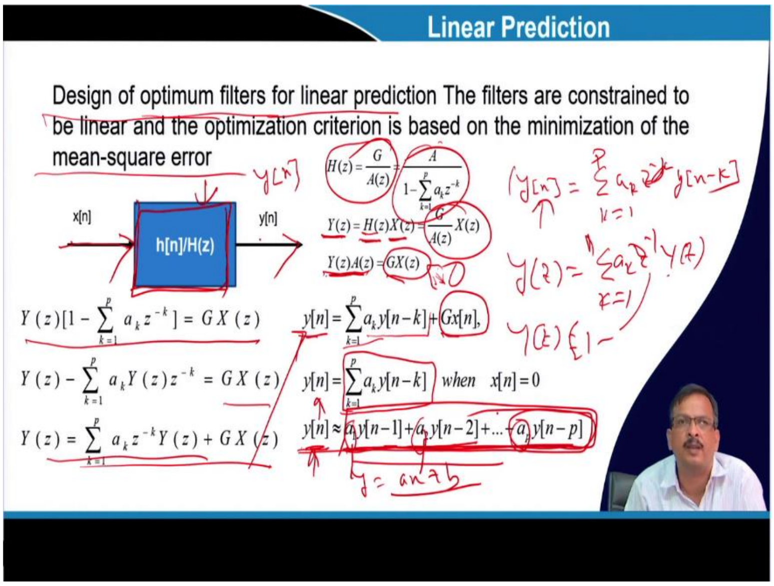
So, I say that the current sample y[n] depends on the previous, current output and the previous q number of output plus present input. So, it is called the autoregressive method, which is why it is written as H(z) equal to G by A(z). So, it only has a pole, no zero. Now, there is an auto-regressive and moving average model, the ARMA model, which has pole zero.

So, I can design and wrap the filter by using some modelling technique; instead of a constraint specification of the filter, I can use a sum modelling technique, and I can design that filter. For example, suppose I said I have given you a signal that looks like there is some variation; local variation is there in the signal.

I want to remove this local variation. What do I do? I can say that local variation can be done by modelling a moving average filter, which is nothing but a low-pass filter. So, I want to smooth that variation, which is nothing but a low-pass filter. So, instead of designing a low pass filter providing ωp, ωs and all kinds of constants, I just use the moving average model, and I just implement the moving average model and smooth the signal.

So, when I say that in a random process, if I want to find out the estimation of the frequency, many times we use the modelling.

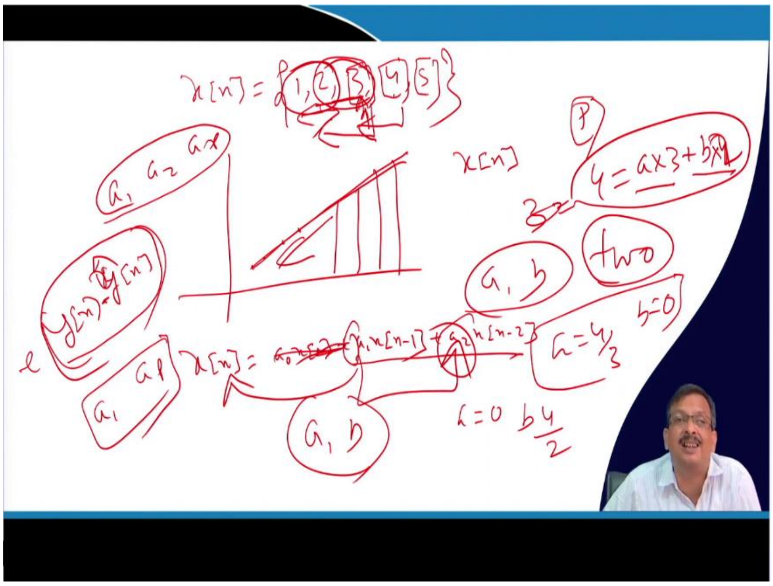
(Refer Slide Time: 09:37)



So, linear prediction is such a modelling example. So, what is linear prediction? Fill about the optimum filter, design linear prediction, and forget about the slides. So, what is a linear prediction? So, what is prediction?

Prediction means that based on the previous data, I can predict the present data and weather forecasting; based on the past experience, the condition of the past experience can predict what will happen if this condition is perceived in future, so that is prediction.

(Refer Slide Time: 10:18)



So, suppose I have a signal x[n], let us x[n] 1, 2, 3, 4, 5 like that.

Now, can I say that I want to predict this sample from the previous two samples? How do I want to predict? This prediction is not non-linear; when I say prediction, a neural network is also a prediction, but it is non-linear.

Here, I said linear prediction; that means I said that 3 is a linear combination of the previous two samples; similarly, 4 is a linear combination of the previous two samples; 5 is a linear combination of the previous two samples. So I can predict the present sample from the future sample, so prediction. So, I can give an example. So, why is this linear prediction possible? Is linear combinational prediction possible? Because you know, let us say this is a straight line.

So, if I know this point, this point, this point, or I know the point, I can draw a straight line, and I can predict any point in the straight line from the previous point. So, I can say, let us say I want to predict x[n] from the previous p number of samples. So, I can say x[n] is a linear combination of the previous p number of samples, be that a 0 into x 0 plus a 1 into x[n-1] plus a 2 into x[n-2] like that.

Or let us say 0 I not consider, let us say 1 k is varied from 1 a 1 x[n] minus. So, from the previous sample, I am predicting the current sample. So, it is a linear combination, and a 1 and a 2 are the coefficients. So, how do I estimate the coefficient? So, when I say my prediction is correct? That is why I chose the design of an optimum filter for a linear prediction. So, once I say it is a linear prediction, it is also a filter

I said the current sample y[n] is a combination of the previous few samples; let us say k is equal to 1 to p ak z^{-k} minus y[n] minus k, instead of the time domain. So, for the previous k p number of samples, I want to take the predicted current sample. So, if I say that, then it is in transfer function will be nothing, but a y(z) is equal to k equal to 1 to p ak z^{-k} y(z).

So, I can say if I take this side. So, it is y(z), which is nothing but a 1 minus this one. And so, when I say transfer function, it is nothing but a 1 by this z. So, it is nothing but an auto-regressive model. So, let us say this is my system, which is called linear predictor H z; H(z) is a system that is the linear predictor.

If I give an x[n] as an input, I can get a y[n] as an output, ok. Now, I said the purpose of this prediction is to predict the current sample from the previous experiment and previous experience. So, if I say that I apply x[n] and I produce y[n], I can say that y(z) is equal to

H(z) and X(z). So, what is H(z)? A by this one. So, I can say y(z) A(z) is equal to G into X(z).

Let us say there is a no input is there; no input there; then I can say y(z) A(z) is equal to 0, or I can if I A(z) if I put it here A(z), value of A(z) 1 minus k. So, I can say y[n]. So, I put the value of H(z) here, okay? Then, once I get this one, this is the y[n]. So, y[n] is equal to a linear combination of the previous p number of samples plus current input.

Let us say the current input is 0; then I can say the current sample y[n] is nothing but a linear combination of the previous sample. So, y[n] is equal to a 1 y[n-1] a 2 y[n-2] plus dot a p minus n minus p. This is a linear equation, y equal to an x plus b is a linear equation. So, that is why I can say then linear y[n] can be predicted or y[n] can be generated from the linear combination of the previous input, ok.

I know, but if you see this is the single equation, and there is a p number of unknown a 1, a 2; sample I know, y[n-1] I know, y[n-2] I know, y[n-p] I know, but I do not know a 1, a 2, a p. Let us say give one example let us say; let us say I said 3 is a combination of let us say or 4 is a combination of, 4 is a combination of a into 3 plus b into 4.

Let us say, sorry, a into 3 plus b into 2, let us say, b into 2. So, the previous two samples, 4, is a combination of the previous two samples, an into 3 plus b into 2. Now, what should be the value of a and b? If I want to find out the value of a and b, I require two unknowns; I require at least two equations, but there is only one equation.

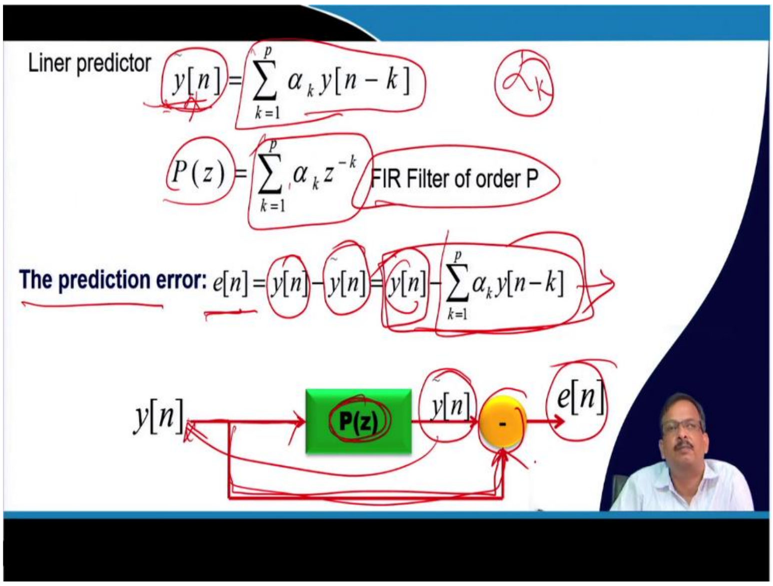
So, if an infinite set of a b solution is possible, a and b can be infinite sets of solutions. Let us say an equal to an equal to 4 by 3 and b equal to 0 can be a solution, or an equal to 0, b is equal to 4 by 2 can be a solution, an equal to something fraction, so many solutions are possible. Then how do I compute this a 1, a 2, a p?

So, if I say let us say a 1 a p is an initial value, and I calculate the predicted value y[n] cap, which is the predicted value. Then what is the error? So, I know present y[n-p] redicted y[n] is my error. So I can minimize that error. So, I want to choose the coefficient that will give me the minimum error. So, the coefficient of the linear prediction filter a 1 to a p can be found by minimizing the prediction error.

So, that is why it is written like this: the filter is constrained to be linear, the optimization criteria are based on minimising the mean square error, and a 1, a 2 is nothing but a filter coefficient. So, instead of providing ωp, ωs, ε, δ all those things pass band edge frequency, stopband edge frequency, ripple, and transition bandwidth; instead of doing that, can I predict, if I say my system is a linear system, that means I can the present output can be predicted linearly from the first output.

Then, I can find out the filter coefficient using estimation theory from the data, which is called modelling. So, when I say auto-regressive modelling, I use it for linear prediction, ok.

(Refer Slide Time: 19:33)



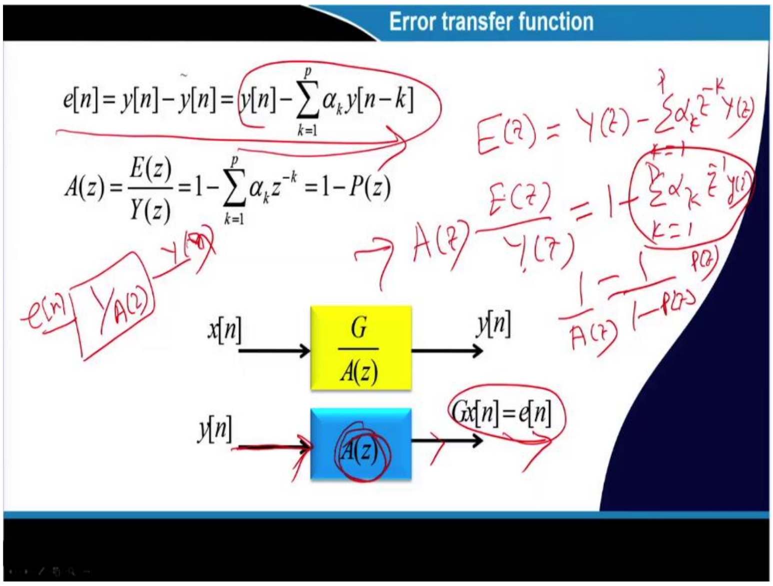
So, what is my world? In the case of linear prediction, what is the problem? The problem statement is I am estimating the present sample from the previous p number of samples, a linear combination of the p number of samples with a multiplication coefficient αk.

So, αk is my predicted coefficient p with the linear combination of the previous output, and I get the present output, ok. So, I can say y[n] if the predicted signal, then P(z) this one is in the prediction filter, which is nothing but an FIR filter; it is nothing but an FIR filter summation of all previous combinations of all previous samples. So, it is a moving average filter or FIR filter.

So, if I say what is the prediction error, e(n) is the current sample minus the estimated sample. So, I can say y[n] minus this one is the prediction error, so if I told you to draw the prediction in a system diagram, I put y[n] as an input pass through the P(z). P(z) is nothing but an estimator that estimates or predicts that y[n] from the previous sample.

Then I know y n; I subtract them, and I get the prediction error. Is it clear? So, a prediction error is nothing but an original value minus the predicted value. So, the predicted value is this one, and the original value is y 1. So, I can say that if P(z) is a prediction filter, which is the linear combination of the previous output, then y[n] minus the estimated signal is my error, no problem.

(Refer Slide Time: 21:36)



Then y[n] minus this one is my error, I know, ok.

So, now, if I take the z transform of this equation, let us take the z transform. So, I can say E(z) is equal to y(z) minus k equal to 1 to p αk z^{-k} y(z). So, if I say E(z) by Y z, y(z) is the input; what I said, y(z) is the input, y(z) is the input, P(z) is the prediction error filter. So, I can say that y(z) by E(z), which is nothing but a transfer function A(z).

So, I said I designed A(z) and applied input y n; I got the output as the error, the mean square error, and the output as the error. So, E(z) by y(z) is nothing, but a. So, I can say

$$A(z) = \frac{E(z)}{Y(z)} = 1 - \sum_{k=1}^{p} \alpha_k z^{-k}$$

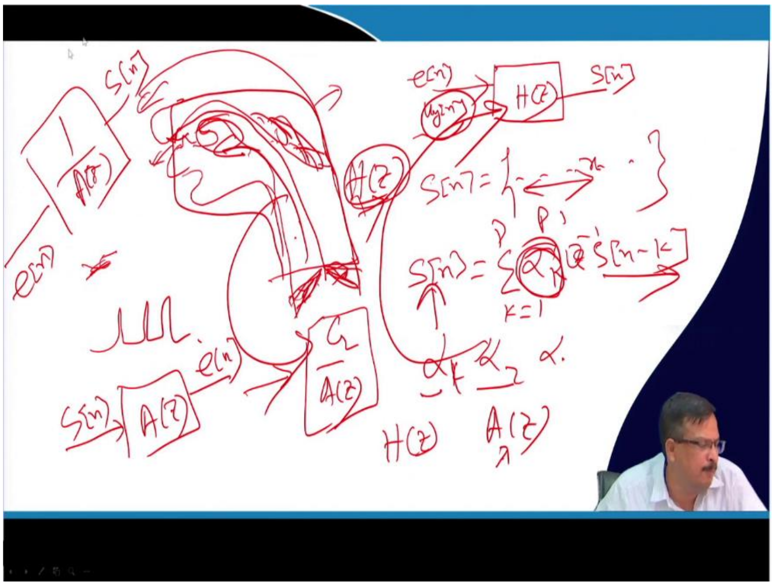
So, if I apply a signal and A(z) is in my estimator or error estimated transfer function, then output is my error.

Now, if I say what is 1 by A(z)? 1 by A(z) is nothing, but a y(z) by E(z); 1 by A(z) is nothing, but a y sorry 1 by A(z) is nothing inverse of A(z), which is nothing but a 1 by 1 minus P(z), this is P(z). Now, if I say, if I input my error e(n) and this is 1 by A(z), I should get y(z), or I should get A(z), I should get y[n].

So, if I apply an estimated error through a one because it is a linear filter, 1 by A(z) is also valid. So, 1 by A(z), I can get y[n] back. So, what is the application? What is the application of this linear prediction filter? Let us say speech signal. So, if you see when I say the speech signal, and if you remember that, what is the speech signal?

Let us give one example of a speech signal: what is a speech signal? Now, when I say speech signal, speech is produced by a human vocal tract. So, what is human vocal tract constraint?

(Refer Slide Time: 24:47)



So, let us say this is the dental part, the upper palate, and the lower palate tongue, and then this kind of system is there, called velum.

This is called nasal passage; this is a nasal passage, this is an oral passage, this is teeth, and then there is a velum will be closed or so that velum either velum will be closed or velum will be. So, either the velum will be closed, or the velum be here. So, the velum may be closed or may open the nasal passage, and then it goes there, and there will be a vocal cord here, there will be vocal cords here.

So, this is the tongue, ok? So, our tongue is fixed at the back side; some of the front side is open. When I produce a sound, these vocal cords are nothing but a membrane; if you see the sun eye, the sun eye in the head, there is a membrane. So, the membrane will create vibration, and when that vibration passes through a tube, the vibration characteristics will be changed depending on the tube transfer function.

So, we will move our tongues in different positions to produce different speech sounds during the production of the speech. So, when I produce different speech sounds, the different speech sounds are only characterized by the transfer function of the vocal tract. The transfer function of the vocal tract is H(z), so H(z) is responsible for producing the different sounds, and for different sounds, H(z) is different.

And this is nothing but an excitation, which is nothing but a source. That is why I say a human speech production system is nothing but a source filter model. So, this is called H(z), which is the vocal tract, and this is excitation, which is nothing but an e(n) impulse; excitation is nothing but an impulse or glottal impulse, which we call glottal impulse.

So, it is written as u g n. So, this produces that impulse. So, this produces an impulse that impulse passes through a filter and produces speech signal S n. Now, it is shown that it can be when you derive the transfer function of the vocal tract; it is shown that yes, linear prediction is possible; it is in the form of G by A(z).

When I produced the tube modelling of the vocal tract, it was nothing but a G by A(z); for details, you can go through my YouTube video; the course name is Introduction to Speech Processing, where this is derived, so G by A(z).

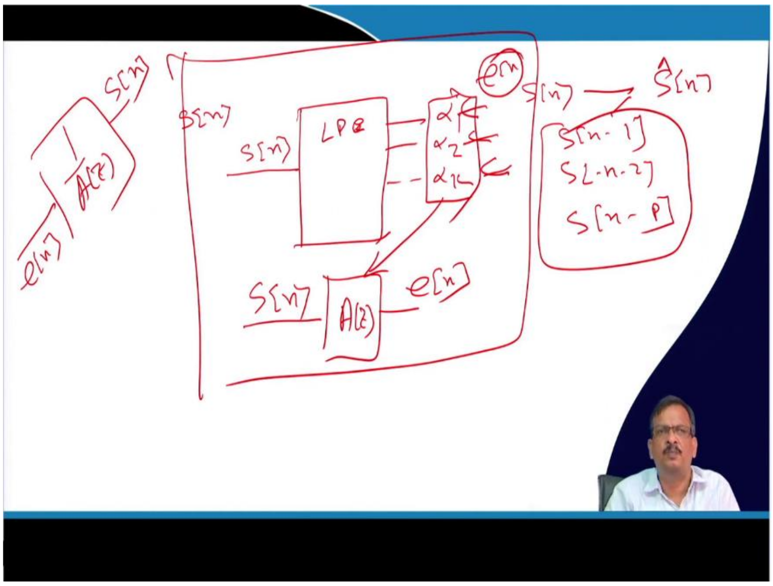
So, I can say the linear prediction of the speech is possible. So, what is the meaning? The meaning is that if I know the speech sample, let us say the speech sample is known for a particular time, speech samples are known; then I can say that any sample can be predicted from the previous sample. Let us say any sample can be predicted from the number of previous samples.

So, I can say any sample S[n] is a combination of k equal to 1 to p αk z^{-k}, or I can say S[n-k]. So, based on the previous number of samples, I can predict the current sample. So, I can estimate that αk; I know the current sample, I know the previous sample, so I can apply a theory, which is called linear prediction theory, to solve for αk.

So, once I know αk, that means α 1, α 2, α 3, so those are characterized this A(z). What does it mean? What if I know α 1, α 2, α, so I can design H(z), or I can design A(z). So, once I design A(z), that means if this is my A(z) if I pass the speech signal S[n] in here, output I get e(n) error, the error is nothing but a glottal impulse. Now, if I design 1 by A(z) and apply e(n), I should get S[n] back, and the speech signal should be back.

So, when I say LPC analysis in the case of speech, there are two things: one is LPC analysis, and one is called LPC synthesis.

(Refer Slide Time: 30:07)



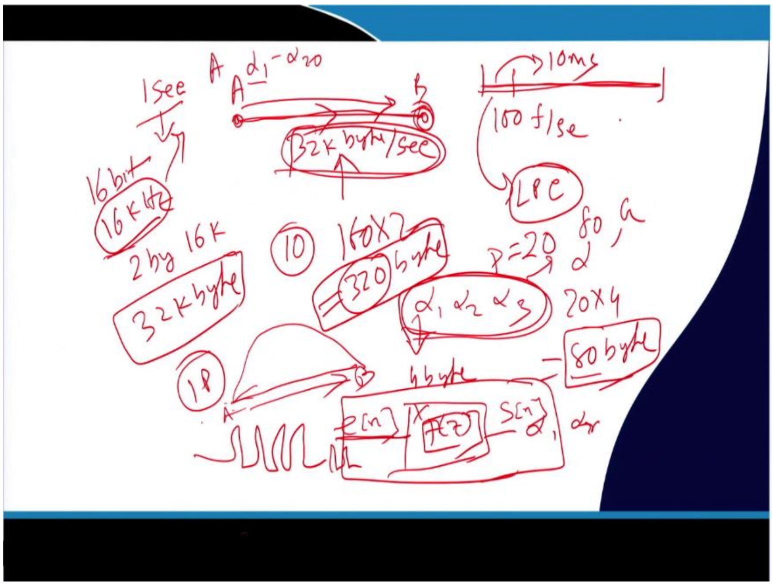
So, what is LPC analysis? So, when I say LPC analysis, that means I know the speech signal; this is my LPC analysis, LPC linear prediction predictive analysis. So, what is my purpose? My purpose is input is my S n; I have to predict this α 1, α 2 dot dot αk that I can predict α 1, α 2.

So, I have to compute those values. How do I compute that value? Minimizing the error: so I know S[n], I know S[n-1], I know S[n-2], I know S[n-p]. So, previous p sample I know. So, from that previous p sample, I will estimate another current speech signal and subtract it, get the error, and minimise the error to estimate the value of α 1, α 2, αk.

Let us estimate what I will do: I will design an error filter, I pass S[n] using those α values, and I get e(n) error filter, so the error signal is nothing but a glottal excitation. Now, this is LPC analysis; I now have my LPC synthesis. So, what is LPC synthesis? In the synthesis part, we will design 1 by A(z) and apply e(n) excitation; I should get the S[n] speech signal back. So, where is it used?

Now, suppose I give an example; suppose earlier before the speech coding; have you heard about the speech coding you heard speech coding before the speech coding came?

(Refer Slide Time: 32:08)



Let us say I want to transmit a 1-second speech data from point A to point B. Now, that is, this speech is recorded with 16-bit encoding with 16 kilo hertz, let us say.

So, 16-bit encoding with 16 kilohertz; so how much memory? So, in 1 second, how much data is generated to buy it into 16k? So, I can say 32 kilobytes is the generated data in 1 second. So, if I want to transfer real-time A to B, the minimum bandwidth is nothing but 32 kilobytes per second, 32 kilobytes per second. I require that much speed, okay or not.

So, now, that is why when I say the voiceover IP and telephone channel, when the telephone channel is discovered voiceover I, today we can make a Skype meeting, which is called voice going through IP. So, there are two kindS[n]etworks: an IP network and circuit switching, which means a dedicated channel is created from the talker to the listeners. That is called circuit switching, but it is an IP network, which means packet switching.

So, in that case, I do not know how the packet will transferred from A to B, but in the case of telephone, there is a duplex communication between A and B. So, if you are a communication student, you know that. Now, voiceover IP will be very cheap because it is shared by many people; but I cannot be guaranteed that this 32 kilobyte per second will be reached at B; unless it is reached, I cannot real-time communication is not possible. Then what is the drawback?

They asked why I should transmit 32 kilobytes; am I not compressing the data? So, how do you compress it? Like the image compression. So, voice compression is voice coding is come. So, how do I compress it? So, let us say in 1 second, my frame rate is 100 frames per second; that means I am analyzing 100 frames in a second, ok? So, for each frame, I compute the LPC coefficient, let us say.

So, let us say I am computing the 20th-order LPC coefficient. So, 20, so p is equal to 20. Let us say that each of the coefficients α 1, α 2, α 3 can be coded at 4 bytes. So, how much memory is required to store the 20 coefficients? So, 20 into 4, so I require 80 bytes. Now, instead of transmitting a sample in 10 milliseconds, if I say 100 frames per second, it means each frame is 10 milliseconds.

So, how much size if I transmit all samples of 10 milliseconds; the size is 10 milliseconds, how many samples will be there? If it is a 16 kilohertz sampling rate, 160 samples will be there; each sample is 2 bytes, so the total memory requirement is 160 divided into 2 by 320 bytes. So, instead of transmitting 320 bytes, I want to transmit only 80 bytes.

So, I implemented LPC analysis at point A, extracted α 1 to α 20, and transmitted those signals to B. Now, on the B side, I will implement 1 by A(z). So, I know α 1, α 20, so I can implement 1 by A(z), and I can be excited by an e(n). What is e(n)? e(n) is a glottal excitation vibration, nothing but an impulse. So, I am excited that 1 by A(z) by an impulse strain, I get back my speech signal S[n], that is, LPC synthesis.

So, I get back my speech signal. So, instead of transmitting 320 bytes, I am transmitting only 80 bytes. So, that much compression I receive. Now, there are a lot of complications that will be there; then there is an f 0, there will be a gain, there will be an impulse position to make it more generalized, or we can say a more natural sounding voice at S[n]; there is a lot of drawback due to this design.

So, those design constraints can be improved by improving the size; instead of only transmitting α 1, α 3, I can transmit along with the f 0 value and along with the gain value and whether it is voice or unvoiced.

So, I can get back all that information, I can design that LPC synthesis, and I get back the data. So, this is one of the uses of linear prediction; there are many applications of linear prediction. So, I can predict it.

So, instead of designing a filter, I am predicting all. So, the filter is a linear transfer function, and prediction is also a transfer function. So, instead of specifying ωp, ωs, bandwidth, transition bandwidth, all those things, I am estimating a certain coefficient value and based on that coefficient value, I am generating the signal again. That is why it is called linear estimation. In the next lecture, I will show you how to calculate α 1, α 2 and α values, ok.

Thank you.