Digital Voice and Picture Communication

Prof. S. Sengupta

Department of Electronics and Communication Engineering

Indian Institute of Technology, Kharagpur

Lecture - 10

Linear Prediction of Speech

.................A new topic and that is linear prediction of speech. And this is what we are going to use in the linear predictive coding or what is known in very short form as the LPC. Now we already have some idea about this topic with reference to our last lecture because our last lecture we had seen, the adaptive differential pulse code modulation technique and there essentially what we did was that we had predicted a given signal x of n, a given sample x of n we were predicting from the past samples that is x(n) minus 1 which is immediate past and also from the oldest samples like x(n) minus 2 etc etc up to x(n) minus p where p is something that we are taking to be the contributions of the past pixels. Means we are considering that up to p samples we are considering there.

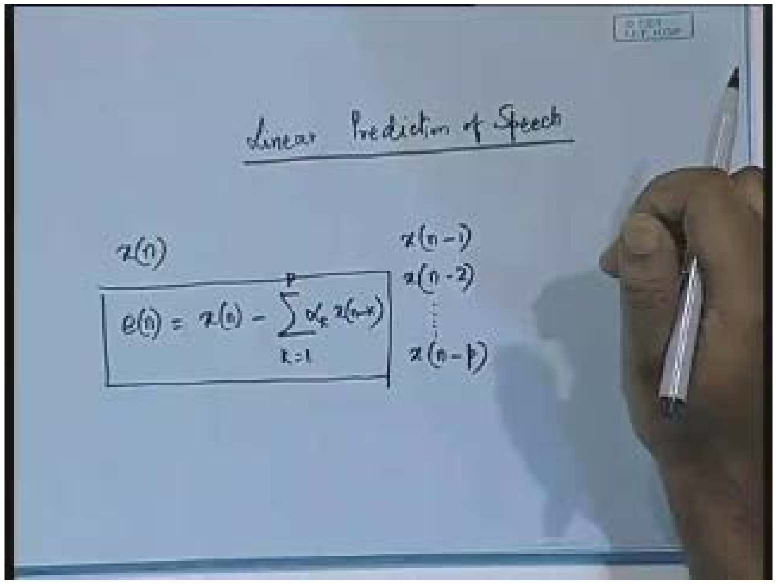
So essentially in the case of ADPCM what our idea was to predict this x of n from all these past samples which are known to us and then we were computing the prediction error. What we were doing was to compute e of n that is the prediction error which was x of n minus the summation k is equal to 1 to p alpha k into x(n)minus k. And then when we were discussing about the adaptive prediction, we were saying that it is this alpha k's which we have to predict which we have to adapt so that this error e(n) becomes minimized. This is what we had seen, and there the whole objective of the exercise that we did in the case of the ADPCM was to use it in the differential coding technique where the major objective was to reduce the bit rate; because as compared to the normal PCN we had seen that how the different techniques really improved upon the bit rates, the aspect of the differential coding itself and then followed by the adaptive differential pulse code modulation which was considered to be the better among all these schemes; the best amongst all the schemes that we had talked of in our earlier lecture. There the basic objective was only the bit rate reduction.

But now in this lecture as well as in some of the subsequent ones where we are taking this topic of linear prediction of speech and going to eventually discuss the LPC vocoder realizations and all that there this error minimization or the adaption of this coefficients alpha k's we will ultimately try to relate these things to the speech model. So if we can model it with respect to the speech then that is what we are essentially looking for.

You see, so far we have done two different analyses and may be it might have appeared to you to be a disjointed analysis. one is where we were considering the vocal tract model and then we were discussing about the formant frequencies, the resonant model corresponding to the vocal tract model and that is one aspect that we have discussed and then we had gone over to the waveform coding and in waveform coding the last that we talked about was the ADPCM which was basically the linear prediction making use of the past samples.

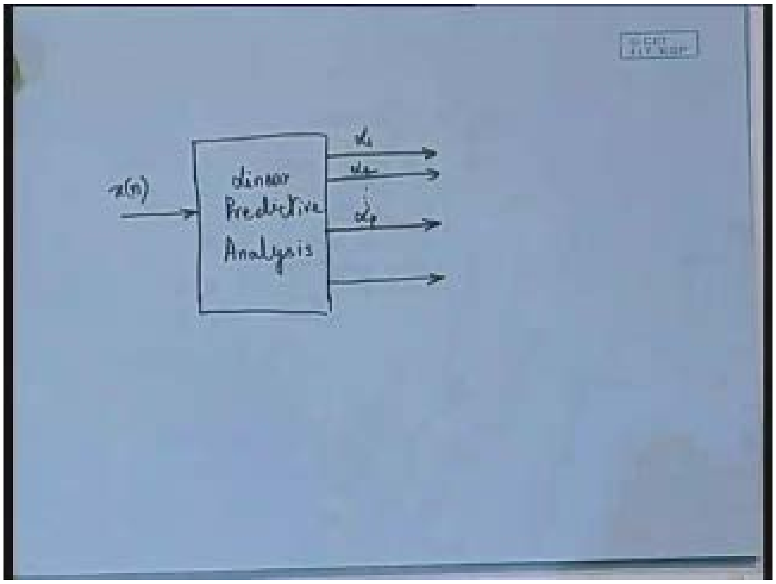
So essentially whatever we had learnt in the ADPCM, it is going to be extension of this idea so that we can ultimately relate it to the speech production model. Hence, now we would like to relate what we had learnt first and what we had learnt only up to the last class. So we will relate these two aspects that is to say the formant frequency model which was essentially a frequency domain model and then in last class what we had discussed was essentially a time domain model because the linear production of speech (Refer Slide Time: 6:15) this equation itself tells us about a time domain model of error analysis and we would like to relate these two.

(Refer Slide Time: 6:22)



First of all, that if this is going to be our model that is to say the error e(n) computed from x of n and x of n minus k. Then considering the adaptive coefficient or adaptive coefficients alpha k’s because if you want to adapt the alpha k’s in that case a block diagram model that would come our mind it something like this that we will be having x of n that is to say the samples as the input and then it will be the input to a block which we are going to call as the linear predictive analysis. And what this analysis block is going to give us as outputs? It is going to give us alpha 1 alpha 2 etc up to alpha P. Because in this case in this equation (Refer Slide Time: 7:27) we have considered up to P terms that means to say the up to P previous samples so it goes up alpha P; and then also we are going to compute the errors which is going to be e of n.

(Refer Slide Time: 7:44)

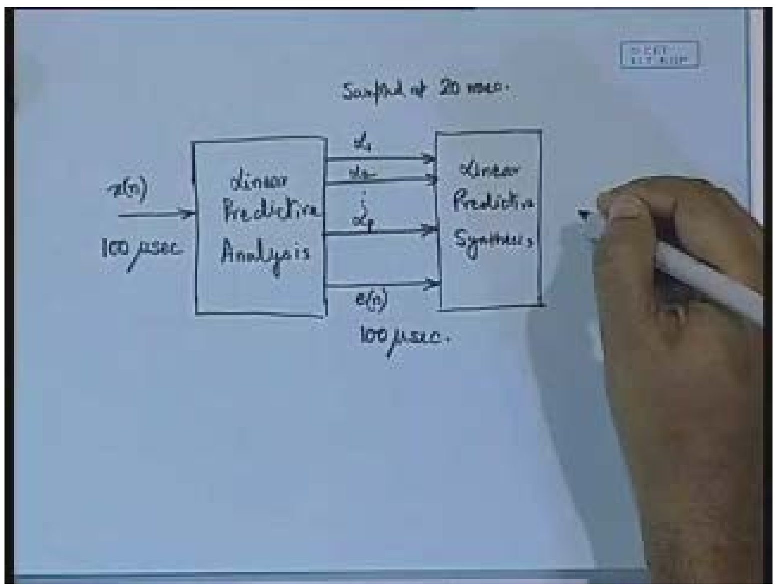


Now, out of all these things alpha 1 to alpha p and e of n which will essentially be the analysis parameters, this alpha 1 to alpha p we had said that these are going to be slowly varying because if we do the analysis on the speech on actual speech samples we observe that this alpha 1 to alpha p's they change only with the change of the syllables so this is at the syllabic rate and the syllables of human utterances they will be changing at a rate of order of milliseconds, order of 10 to 25 milliseconds that is how the that is the rate at which the syllables may change and then so that means to say that alpha 1 to alpha p they are slowly varying parameters.

So if we write the sampling rate this x of n is going to be sampled at a rate of let us say 10 megahertz 10 kHz sorry I am extremely sorry it is 10 kHz and then it should be sampled at every 100 microseconds whereas this alpha 1 to alpha Ps they can be sampled at a much lower rate; let us say that these are sampled at 20 milliseconds, that is fairly a sufficient rate because less than 20 milliseconds is not going to change so it is a slowly varying parameter alpha 1 to alpha P.

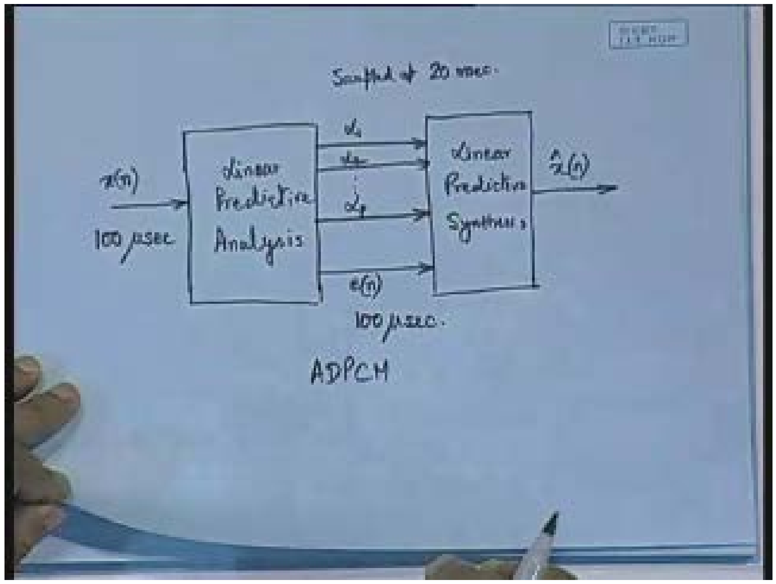
Now if alpha 1 to alpha p's are maintained constant or relatively stable over a time period of 20 milliseconds let us say, in that case the e of n the error n that is what we have to sample; with every sample we must send the error information at least; because what we are gaining is that, instead of sending x of n in to the channel, we are sending this slowly varying parameters alpha 1 to alpha p and then e(n) but e(n) has to be sampled at the same 100 microseconds rate. e of n is changing fast. But this e of n where we are gaining is that the variance. Thus the variance of this e of n is much lower as compared to the variance of this x of n and that is how we are gaining by having the predictive technique. So you learn that G p is greater than 1 that is is to say the prediction gain is greater than 1 so that is how we are gaining in terms of the predictive response.

(Refer Slide Time: 11:03)



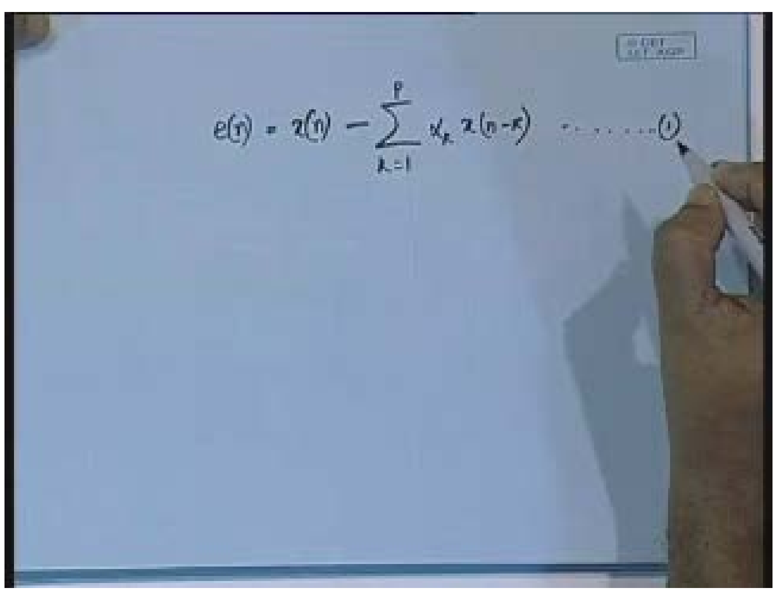
Now this alpha 2 to alpha p as well as this e(n) they will be received at the decoder end. So at the decoder end we must have a corresponding block which will not do the analysis but which will do the synthesis. So this corresponding block at the decoder we must say as linear predictive synthesis. And this linear predictive synthesis is going to generate what we can call as x cap of n. Why x cap is because again this alpha 1 to alpha p these are going to be somewhat our estimates which are based on the speech model and this estimate may not be exact and that is why at the reconstruction x(n) will not be exactly construction so we call it as x cap n. So this is the model that one has to follow in the speech coding based on whatever we had studied for the ADPCM. So up to this it is nothing but the ADPCM what we had studied.

(Refer Slide Time: 11:49)



Now let us go back to our original equation. Our original equation on the error says that e(n) is equal to x(n) minus summation k is equal to 1 to P alpha k x(n) minus k. Let us call this to be our equation number 1.

(Refer Slide Time: 12:14)

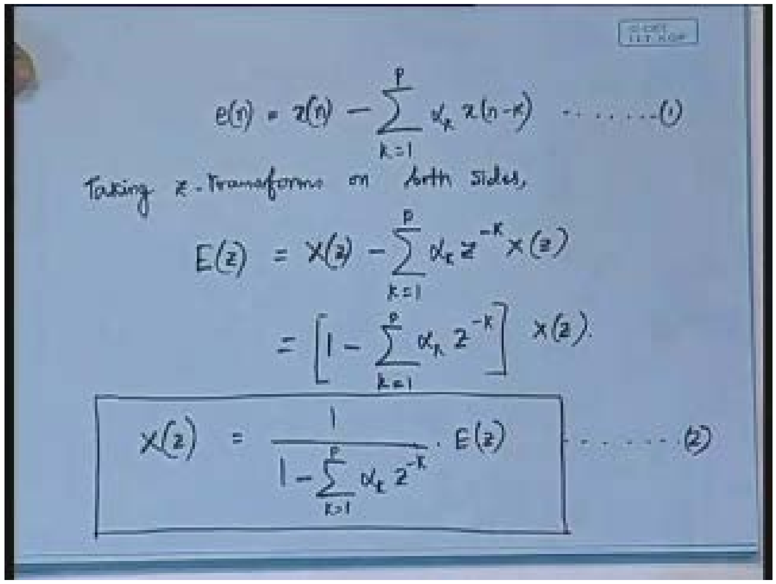


Now we take the z transform of this equation. Both the both the sides we take the z transform. So if we take the z transform then what are we going to get? Taking z transforms on both the sides we are going to get; capital E we are going to write in the z domain, capital E of z should be equal to x of z. Now we are going to write it as capital. This x(n) corresponding z transform form is x of z capital X we write. And what is going to be the next term? Minus......... the summation remains because alpha k's are the individual coefficients so of the coefficients will still be retained in the z transform. So k is equal to 1 to P alpha k this will remain and then what will happen?

[Conversation between Student and Professor – Not audible ((00:13:14 min))] Beautiful; this is going because it is x of n minus k so when we take the z transform of this this is going to be z to the power minus k times x of z. So this we are going to write as alpha k z to the power minus k x of z and this we can write as 1 minus summation k is equal to 1 to p, alpha k z to the power minus k this whole thing multiplied by x of z. So then in that case we can write down; rewrite this equation as; x of z we can write as 1upon 1 minus summation k is equal to 1 to p alpha k z to the power minus k this times E(z). Now this we are calling as equation 2. But let us make a note of this equation.

Nothing great because, after all what great thing we have done; we have only taken the basic error model and then taken the z transform of that, taking this z transform is also not new to us, all of us are familiar with this from our very early years of signals and systems or digital communication and digital signal processing everywhere we have been reading z transforms so it is not something new. But now we remember this equation and let us go over to the speech model what we had discussed few classes back.

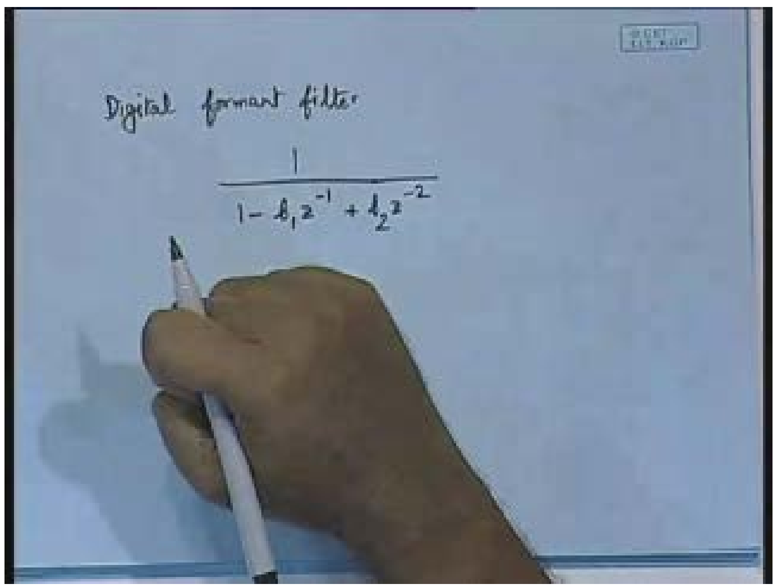
(Refer Slide Time: 15:06)



Now, what we were doing at that time; now here we are beginning with the time domain analysis and we have obtained an equivalent frequency domain analysis for this. Now, in the formant model or in the vocal tract model what we have considered? There, in fact, we had discussed about a frequency domain model. In fact the vocal tract filtering if we are trying to look at, basically what we have? There the input to the vocal tract system is a train of pulses which are serving as the excitation and it is exciting the vocal tract model and there is a vocal tract followed by the radiation at the lips. I mean, there is that cavity, the mouth cavity and with the lip opening when we are finally delivering the speech there we are having what is called as the radiation.

Now these two aspects together actually makes it a two pole filter and in fact every formant frequency or every formant resonance is actually modeled as a cascading of two digital filters and both are digital filters with one pole. So if you are cascading it then you get a two pole model of the digital filter. So, the digital formant filter rather you can model it like this. the digital formant filter, we can in general model its transfer function as 1 upon 1 minus b 1 z to the power minus 1 plus b 2 z to the power minus 2 this is obvious because one pole means it will be 1 minus something let us say 1 minus a 1 or 1 minus a into z to the power minus 1 if you take and then the other if you take as 1 upon 1 minus b into z to the power minus 1 multiply the two; cascading means it will be just the multiplication of two and there if you equate the coefficients you will be finding some coefficients associated with z to the power minus 1 and some coefficients associated with z to the power minus 2; for z to the power minus 2 it is going to be with the positive sign because it is a cascading of two such filters. So this is for every formant we are going to have a digital formant filter like this.

(Refer Slide Time: 18:14)

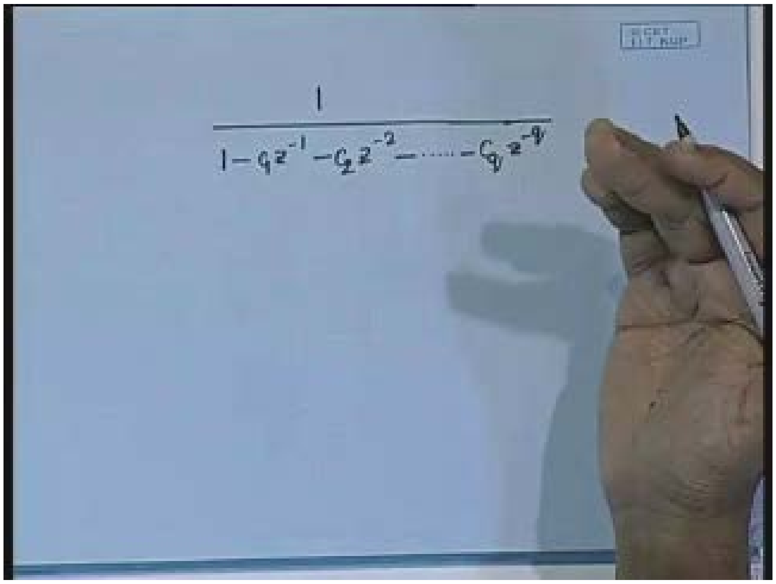


Now what did we say?
We said that there is evidence of at least 3, 4 or up to 5 formant frequencies we have the evidence of their existence in the speech total speech bandwidth; if we take it up to 3.4 kHz or so then we find existence of several such formant frequencies individually which correspond to the resonances.

Now for each of these formants we are going to have such two pole filters. So net effect-wise considering four formants we are going to have totally a combination of eight such things. And then also if you remember, if you if you recollect our discussion sometimes back, considering the radiation model we had said that the radiation really is a frequency dependent term and we have to compensate for that and there we are providing a compensation in the way that we introduce a minus 6 dB per octave kind of a stuff just to compensate for that. So in order to do that we must be having a so the spectral compensation so the minus 6 dB per octave spectral compensation we will be doing by a filter which will be having a transfer function of the form 1 upon 1 minus bz to the power minus 1. So this also will be there. So you will be having compensation which is a first-order filter and then we you will be having for every formant a second-order filter.

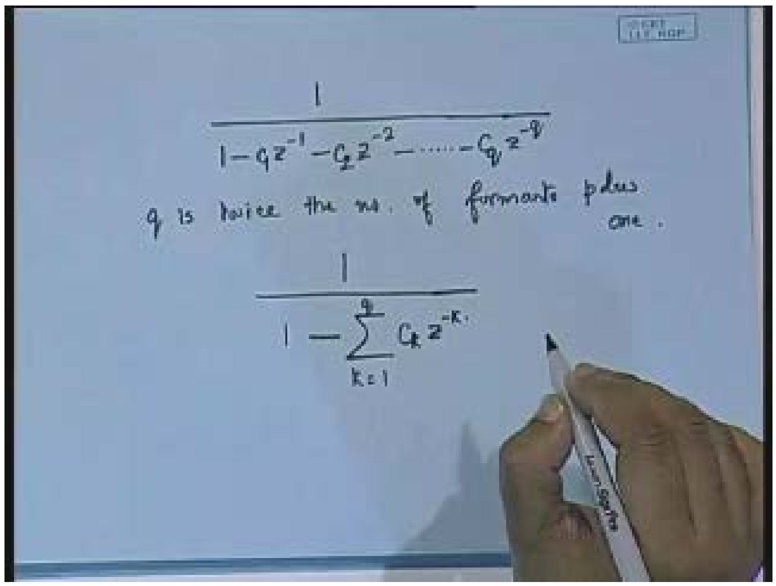
Now supposing you have four formants, in that case totally you are going to have 4 into 2 8 plus 1 realized from this so it will be a ninth order filter. So, the order of the filter is going to be 2 n plus 1 where n is the number of formants and it is 2 n plus 1; plus 1 because of this spectral compensation. Now, 2 n plus 1 is going to be an odd number so we are going to have an odd order filter, totally. So, if we take the product of all these terms and since there are odd number of poles so after the multiplication we are going to get a transfer function which will be of this form: 1 minus a coefficient we call it as c 1 z to the power minus 1 minus c 2 z to the power minus 2 minus and then it goes on minus c q z to the power minus q; all terms are with minus signs. This is what we are going to have.

(Refer Slide Time: 21:39)



Now here q that is to say the order that is what we having, q is twice the number of formants plus 1. Therefore, now in general we can write it; instead of writing like this we can write it in a series form; what is that? We can write it as summation k is equal to 1 to q c k z to the power minus k.

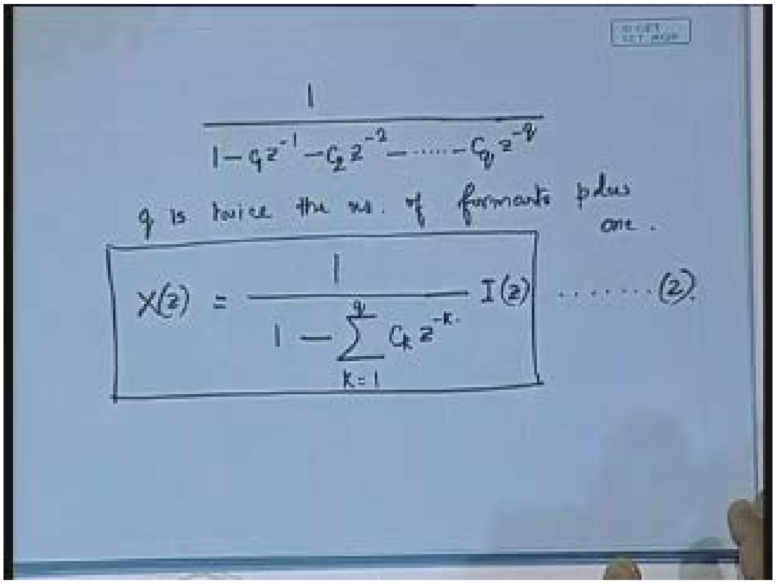
(Refer Slide Time: 22:28)



Then what is the input to the system?

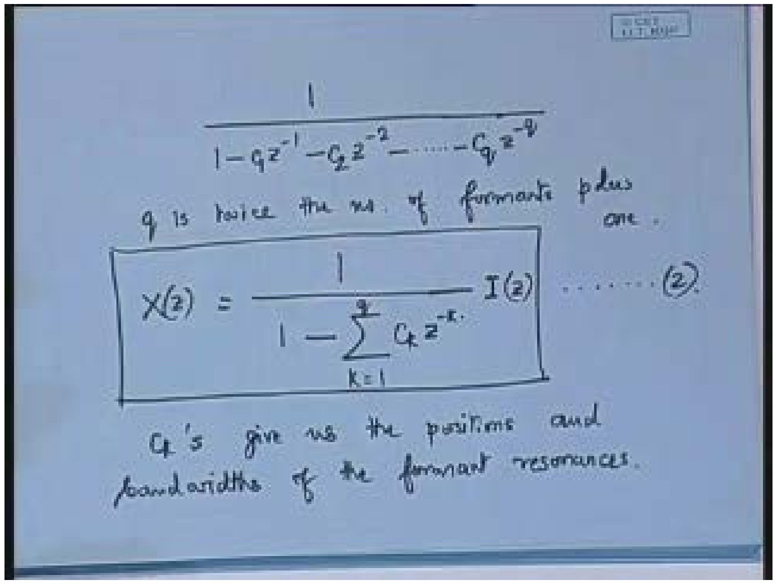
the input to the system is the excitation, the impulses, the excitation what we are producing so we are callings its corresponding z transform to be I of z. So this is z transform of the impulse inputs, impulse to the vocal tract. And what is going to be the output? The output is going to be the x of z. So x of z is related to I of z as 1 upon 1 minus summation summation k is equal to 1 to q c k z to the power minus k. Let us call this as equation number 2. This is also a very important relation. Again nothing new because we are already knowing about the formants and we are already knowing about the digital filters and its poles and everything; nothing great information about it. But something should be striking.

(Refer Slide Time: 23:30)



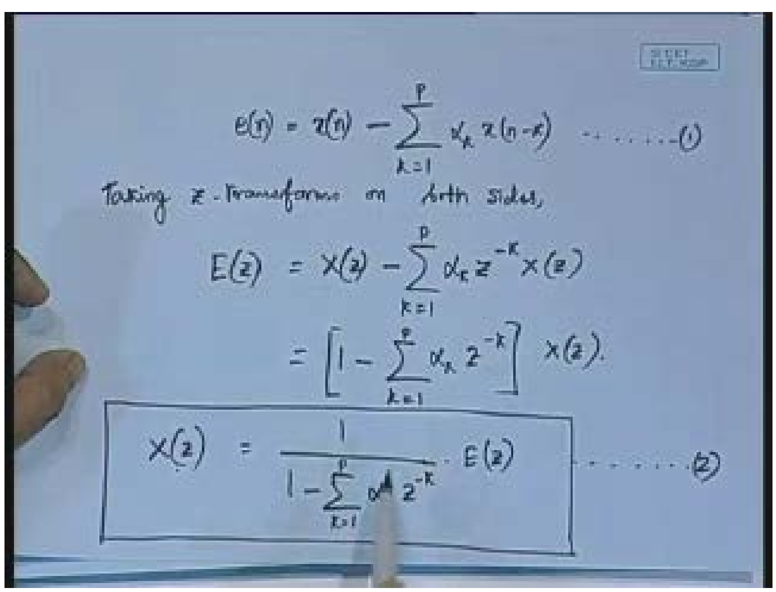
Equation number 2 is relating the x of z with I of z with the speech parameters. Now, after all what are these c k's, what is the physical significance of these c k's? These c k's are giving us the position and bandwidth of the formant resonances. C k's give us the position and bandwidths of formant resonances. So it is definitely related to the speech production model. These c k's essentially give us the speech production model.

(Refer Slide Time: 24:38)



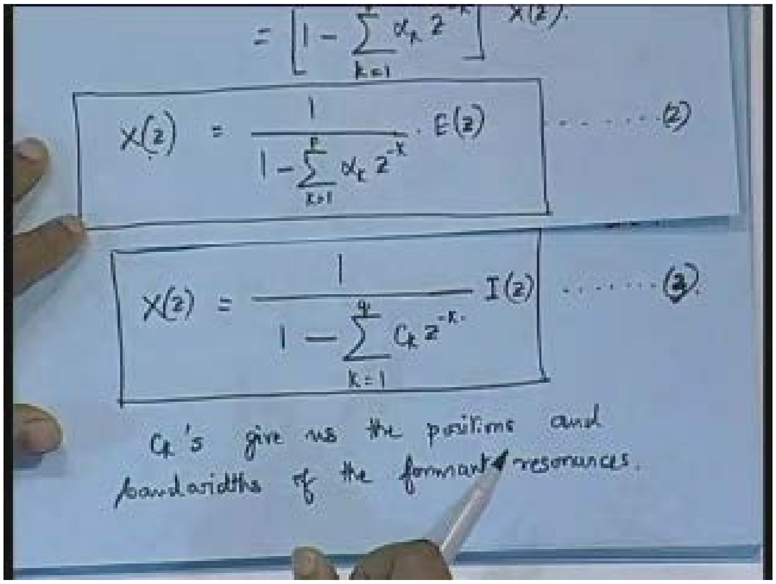
Now let us go back to equation number.......... oh we should call it as equation 3 because we have already numbered the earlier equation as equation 2. And what was equation 2? x of z related to E of z.

(Refer Slide Time: 24:54)



But there is a term, the transfer function is 1 minus summation k is equal to 1 to p alpha k z to the power minus k. Compare it with this.

(Refer Slide Time: 25:08)



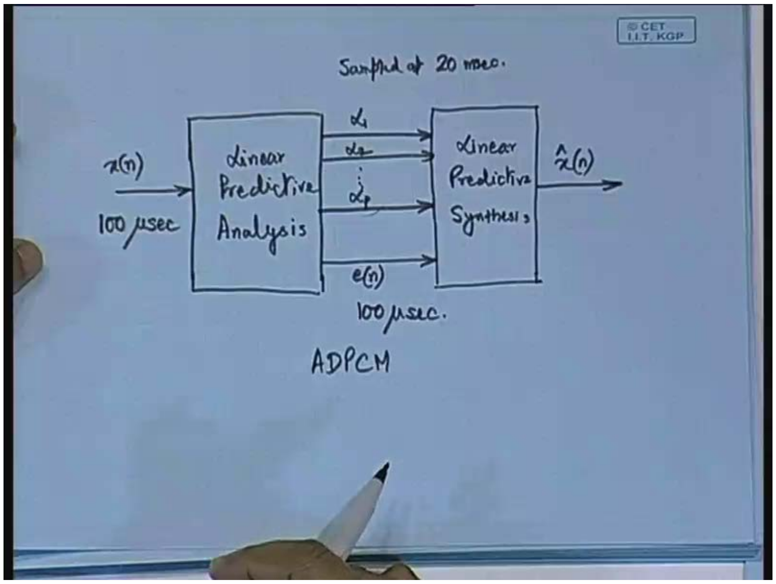
Are not these two equations resembling each other? They are perfectly. Here it is x of z, here also it is x of z, the only thing is that here we have got p number of terms and here we have taken q number of terms. Let us say that we make p to be equal to q. p is after all, as per our choice that up to how many terms we did; let us say that we make p is equal to q means either we choose the number of total number of formants that; up to how many formant resonances are we considering or we consider that? Okay, up to what order of prediction are we having; p could be made by our choice, p could be made equal to q.

And then whatever alpha k's are there in this equation will perform the role of this c k's over here and this error term E(z) is similar to what we have as I of z. Now there lies one big question; what I said about E of z? E of z is...... time domain form is E of n, but I have said that E of n you should sample once in every 100 microseconds, every sample E of n has to be estimated.

Now look at I of z; that means to say that I of z also should we do like that; that okay, for every sample we must......... because after all I of z also should be generated like that. But what is after all I of z? The excitation pulses; the excitation pulses we are producing at with certain pitch and with certain amplitude and the speech and for speech the speech and amplitude they are not varying that fast. These are not the parameters which are varying at 100 microseconds rate, this will be varying at the syllabic rate of 10 to 25 milliseconds. So if we sample the pitch and the amplitude information for every 20 milliseconds for this I of z that should be sufficient for us.

Now that means to say that in our linear predictive analysis we can have the alpha k's or the c k's, this could be definitely sampled at 20 milliseconds of rate between 10 to 25 and also the parameters of I of z which are given by the pitch and the amplitude both we can detect because amplitude can be detected by using big detection techniques and pitch estimation we have already seen; there are different techniques for having pitch estimation. So we can make use of the pitch estimation and amplitude estimation in order to estimate this and this being a slowly varying quantity then in this block diagram where we had considered that this alpha 1 to alpha p should be sampled at 20 milliseconds but n should be sampled at 100 microseconds, now we are going to modify this block diagram and say that in that block you can sample all the parameters at 20 milliseconds of rate and if you can do that then you are going to produce a low rate low bit rate speech coder.

(Refer Slide Time: 00:28:38 min)

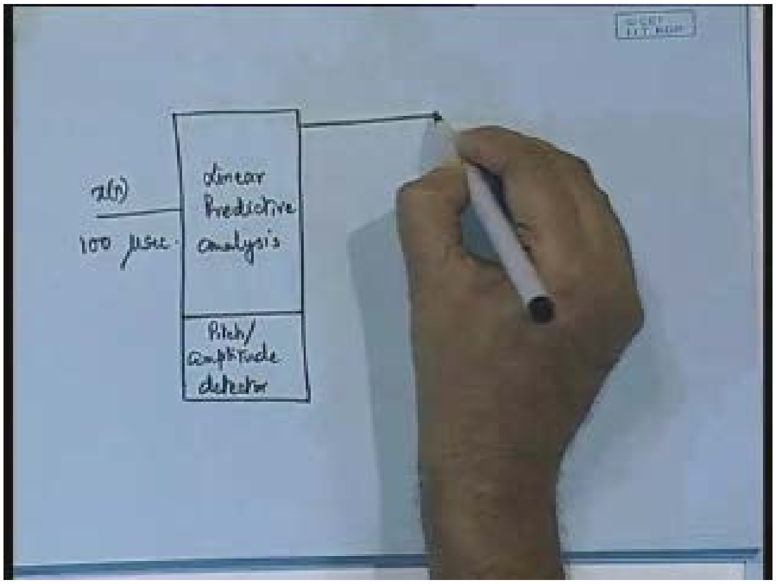


This is also a low rate, why? in the sense that E of n is having much smaller variance; even though it is sampled at 100 microseconds it is having a much smaller variance but alpha 1 to alpha p are already sampled at a very slow rate of 20 milliseconds. But in the modified linear predictive analysis system we are going to have everything, all the outputs of this block, linear predictive analysis block, that can be sampled at a 20 milliseconds of rate and we are going to have a much more efficient speech coder.

So what will be the block diagram model for this?

That should be something like this. So now this is a block that we draw and we separate this block into two parts: one is the linear predictive analysis and below that we are having another sub-block which we call as the pitch oblique amplitude detector and the input to this is going to be the same x of n and this x of n is still sampled at 100 microseconds; if the rate is 10 kHz it is 100 microseconds.

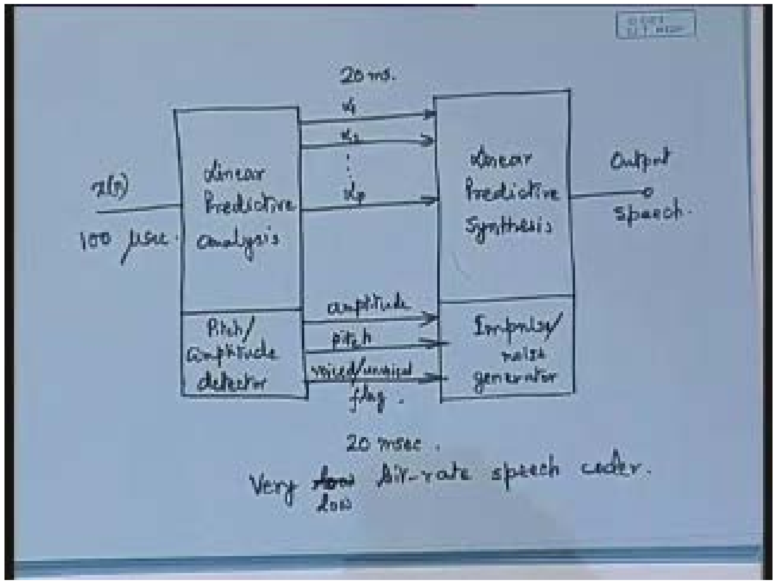
(Refer Slide Time: 30:49)



Now, the linear predictive analysis is definitely going to give us alpha 1 alpha 2 up to alpha p as before and the pitch cum amplitude detector is going to give us the amplitude information; it is going to give us the pitch information and also the voiced/unvoiced flag; we know how to detect all these things, we have already discussed this aspect. So now all these things will be sampled at a much lower rate. These also will be sampled at a rate of 20 milliseconds (Refer Slide Time: 31:36) and this amplitude pitch and this voiced/unvoiced flag they also will be sampled at 20 milliseconds.

So now this will go to the next block and the next block will be to the decoder side, so decoder side we will still call it as the linear predictive synthesis and here whatever amplitude pitch and voiced/unvoiced flag information are obtained, we will use this as an input to an impulse or noise generator. If we know that at what pitch we have to generate the impulses, the period in between the impulses then we will be generating the impulses accordingly. If it is a voiced then we have to generate the impulses with the deciding amplitude and if it is unvoiced in that case the noise generator has to be switched on. So ultimately with this impulse noise generator and all these predictive synthesis, we will be generating the output speech. So this is going to be a very low bit rate speech generation. This is a very low bit rate low bit rate speech coder.

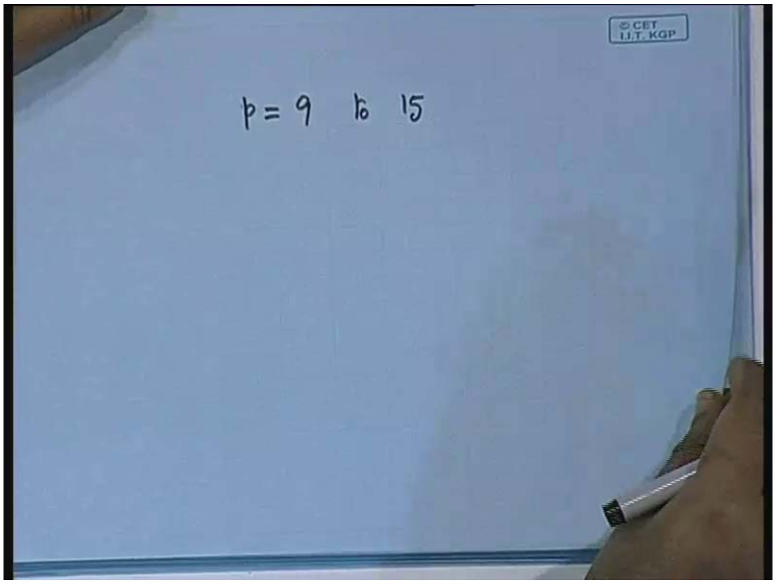
(Refer Slide Time: 33:45)



Why very low bit rate is because everything is at 20 milliseconds of rate; just compare it with the original sampling rate of 100 microseconds, 20 milliseconds is naturally we can afford to call it as a very low bit rate. So this is the basic philosophy.

Therefore, what we have really seen is that the linear prediction that separates out the excitation properties of the source from the vocal tract filter. Now the source parameters can be derived from the error signal and the vocal tract will be represented by the linear predictive coefficients.

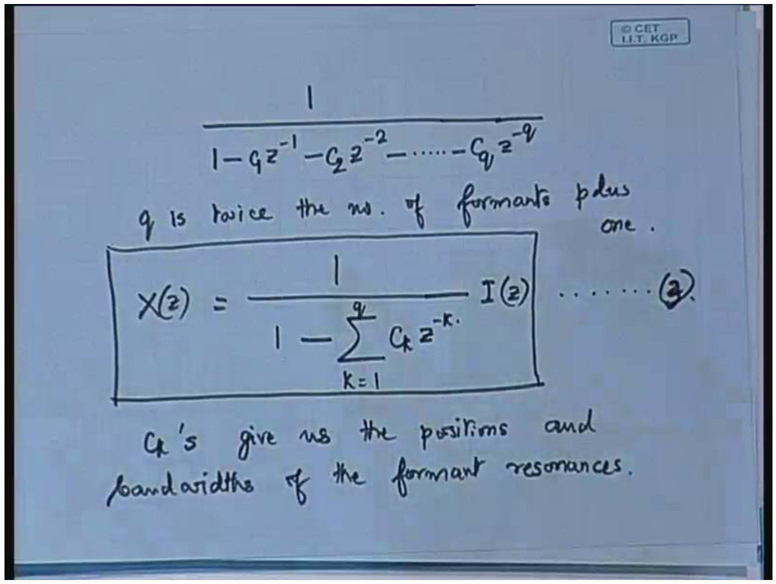
(Refer Slide Time: 00:34:32 min)



Now a practical linear predictive filter that can be, I mean that is that is also having some limited order and that is between............ so the practical values of p will be between 9 to 15. In such a case we are going to have.......... for p is equal to 9 it will be a fourth order so four formant frequencies have to be chosen for p is equal to 9 and for p is equal to 15 seven formant frequencies are to be chosen.

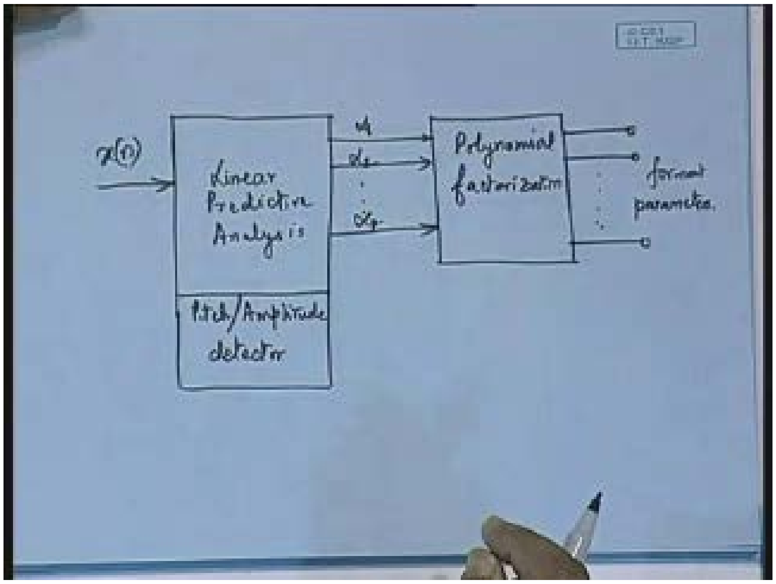
Now so far what we have discussed, there we are sending this alpha 1 to alpha p information directly into the channel, the estimates of these, the prediction coefficients we are sending into the channel directly into the communication channel but actually what we can do is that rather than doing this we can extract the formant frequencies because after all within this alpha 1 to alpha p the formant frequency information is also kept hidden so we can send the formant frequencies which means to say that we have to derive that from this alpha 1 to alpha p's. So we can do a kind of polynomial factorization because after all, look at this denominator term, the denominator term is a polynomial in z.

(Refer Slide Time: 00:36:14 min)



This is a polynomial term and if we do a factorization, if we do a polynomial factorization in that case we should be able to extract the roots. Now the roots will be complex roots which we can get out of the polynomial factorization and after all we have to do the polynomial factorization at a real time; at real time we have do the polynomial factorization so with that we will have some change little change in the block diagram so there we are going to have it as that x of n should be the input and as before we are going to have our block as the linear predictive analysis and the pitch amplitude detector, so this is the linear predictive analysis and here we are going to have pitch oblique amplitude detector and here alpha 1 alpha 2 up to alpha p and then here we have to do what is called as polynomial factorization and this polynomial factorization is going to yield what is called as the formant parameters.

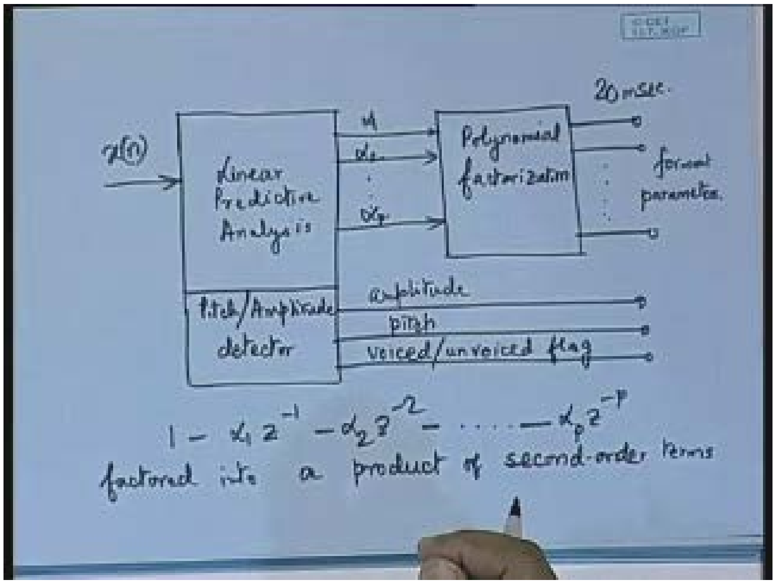
(Refer Slide Time: 38:19)



So the input to this is x of n and the output is going to be the formant parameters and from the pitch amplitude detector we are going to have these three things that is to say the amplitude, then the pitch and then the voiced/unvoiced flag. So, even the formant parameters also will be sampled at the lower rate. All these could be sampled at a rate of 20 milliseconds. So if this also remains as a............

In fact, the formant parameters will be somewhat less as compared to this alpha 1 to alpha p's so there also we will be saving some information. But only thing is that this polynomial factorization that means to say the factorization of this quantity that is 1 minus alpha 1 z to the power minus 1 minus alpha 2 z to the power minus 2 etc up to minus alpha p z to the power minus p this has to be factored into a product of second-order terms. So these are to be factored into a product of second-order terms second-order terms and then we will be able to extract the complex roots after this factorization.

(Refer Slide Time: 39:56)



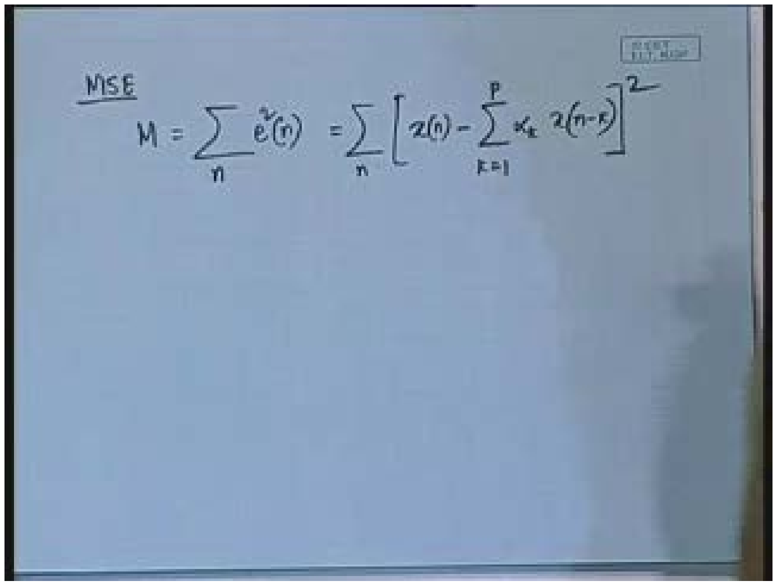
But mind you, the constraint that will be put on us is that this factorization has to be done at real time; real time in the sense that not very back but once in 20 milliseconds, but doing the polynomial factorization once in 20 milliseconds itself is also a very demanding thing, computationally it will be demanding. And let us have a look at another aspect that how good is this alpha 1 to alpha p computation as far as the real time is concerned.

Now we got a feel in the last class when we were discussing about the adaptive prediction aspects with respect to the ADPCM. There, what we had discussed............ so we will briefly focus upon that aspect again just to see that computationally what is going to be involved for determining these coefficients alpha 1 to alpha p. Now there what we had said is that how this alpha 1 to alpha p's should be decided; it should be decided based on the error. The error e(n); not the error e(n) directly but the squared of the error e square n if we take and since it is going to be the mean square error so we should sum up this e square n over certain number of n's and then we have to minimize that quantity. So if we decide on a if we define mean square error which we denote by M and we write it as M is equal to the summation over n e square of n; so this is what we have to really minimize, the mean square error or in short form we write it as MSE.

Now what is this summation e square n?

That is going to be the summation over n x of n minus summation k is equal to 1 to p into alpha k x of n minus k this term square; this is what we as the mean square error.

(Refer Slide Time: 42:39)

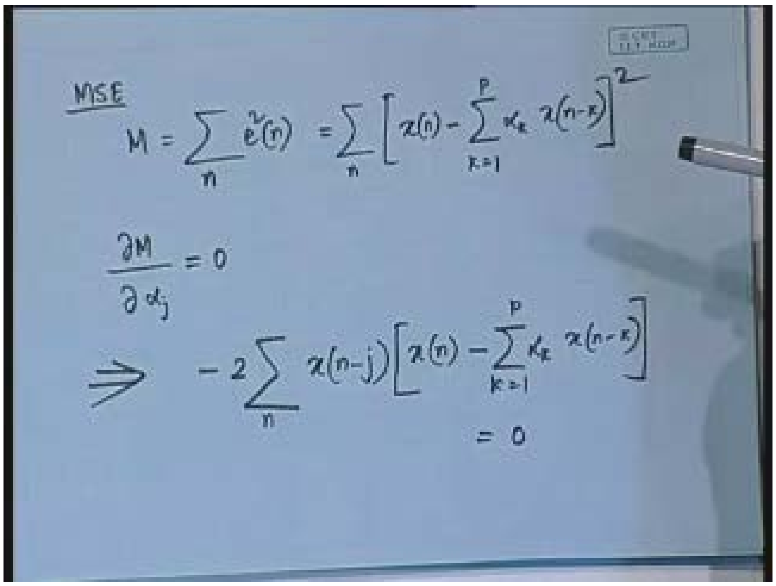


Now we have not put any limit to this n mind you. But actually speaking how to compute this e square n?

We must be computing e square n over finite duration. Now, finite duration means we cannot afford to have any infinite series and try to compute the e square n. So to make it finite what we ultimately have to do is to restrict this M. But we have not put any restriction in terms of this limit in this summation of n so we have kept this limit of n open-ended at the moment; just kept it as n but then we have to see that how are we going to minimize this error and we determine the alpha case accordingly. In principle that is quite simple and we discussed about that also.

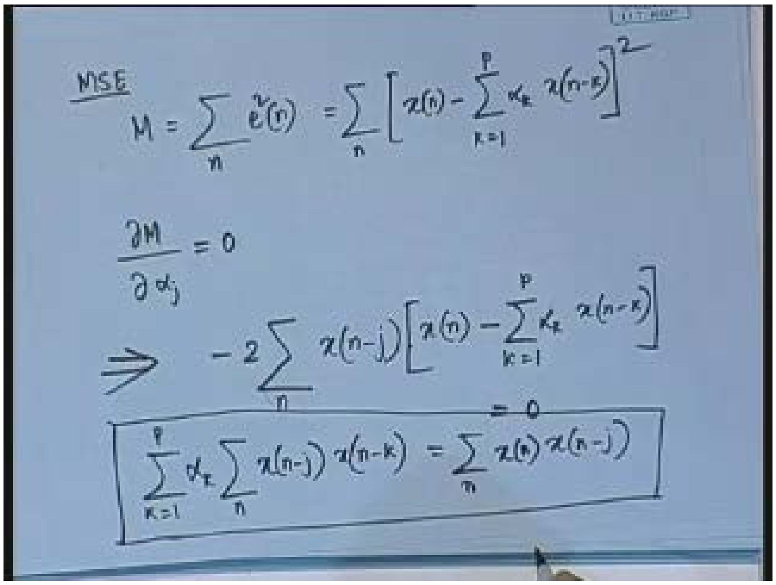
So, if we said this quantity; that is to say if we take this derivative that is to say if we say take delta M delta alpha j that means to say that with respect to some specific alpha we differentiate, so if we take a derivative of delta M delta a j and if we equate to 0 this will mean what; take the derivative so what would you get? Here you will be getting, this is a squared quantity which is written within the brackets which means to say that this will give rise to some minus 2 it will come. In fact what you will get is minus 2 summation over n x of n minus j. You will soon realize that why this x of n minus j comes; and then the quantity that we write within the bracket that is to say x(n) minus summation k is equal to 1 to p alpha k x(n) minus k. This quantity should be equated to 0. This is what will be getting as delta M delta a j delta alpha j.

(Refer Slide Time: 45:11)



Now you see that in this expression whenever we are differentiating with respect to alpha j all other alphas that means to say the terms involving alpha 1 alpha 2 alpha p's everything will be zero except for one term that contributes to alpha j. So there will be only one term in this summation series involving alpha j and then when we take the derivative of this we will get it like this (Refer Slide Time: 45:49) so this will actually contribute to that x of n minus j term; x minus j term will come from that, you can just verify this derivative process it is quite simple so this becomes equal to 0. So if this becomes equal to 0 in that case we obtain that: summation k is equal to 1 to p alpha k into summation over n x of n minus j into x of n minus k and this will be equal to the summation over n x of n into x of n minus j; it follows directly from this equation; summation x minus x(n) minus j x(n) is already there and then we also have this quantity that is to say summation alpha k where k summation is actually from 1 to p but again because of this summation because of this summation of x(n) minus j over n we get a quantity; again this x minus is n minus j here, here it is n minus k so everything put together comes under the summation of this summation over n. So this is one relationship that we are going to obtain.

(Refer Slide Time: 47:15)



Now this equation will be............ how many such equations are we going to have? Not only one equation, but we are going to have p equations because we can write these equations for j is equal to 1, 2,......... up to p so there will be p equations. So, if we now have to solve for............ how many unknowns are there? There are p unknowns; alpha 1 to alpha p's; all of them are unknown quantities so p equations we can compute p unknowns, it is possible to compute that. But that would involve what? That will involve a matrix inversion of; what size matrix? p by p; p by p size matrix has to be inverted again at real time, 20 milliseconds of time we have with us and p by p matrix inversion has to be carried out. So again this is also highly a computationally demanding aspect. So had it not been so, had there been no computational challenge of this kind I could have completed the lecture related to the linear predictive coding in just this class itself. Because this class basically tells you about the nut shell of the predictive process but then the involved things are that how to really solve this equation in a very efficient manner.

People have done lot of researches on this aspect and we will discuss about some of these research aspects and also show you that how efficiently these parameters alpha k’s can be computed. If we know that then at least we can feel confident that yes, we can use this for the computation for ultimately producing a linear predictive coder model, one LPC model for our speech coding process.

So in the subsequent classes we are going to discuss the methodologies for this; the methodologies for the efficient inversion of this p by p matrix or if we can do it in any other alternative mechanism, if the alpha k’s could be extracted a few methodologies have been already suggested in the literature, so we will go through all these things. So for this class let us................... thank you.