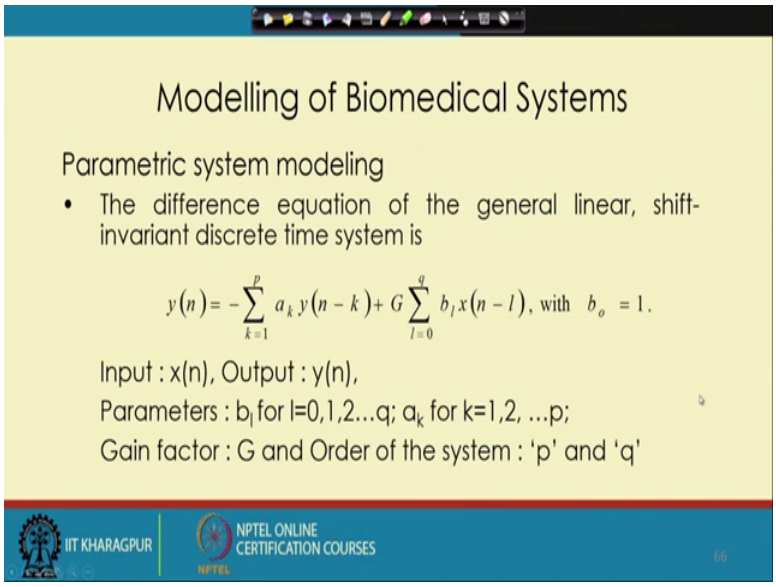
Biomedical Signal Processing
Prof. Sudipta Mukhopadhyay
Department of Electrical and Electronics Communication Engineering
Indian Institute of Technology, Kharagpur

Lecture - 42
Modelling of Biomedical Systems (Contd.)

Now, we will talk about the parametric modelling. So, for the parametric modelling we will look at.

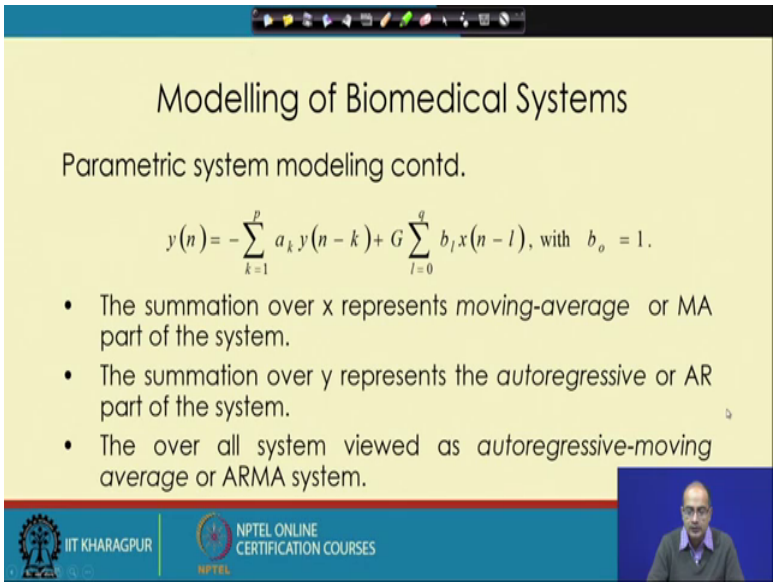
(Refer Slide Time: 00:23)



A difference equation it is a general difference linear equation shift invariant 1. So, we get actually the output y n in terms of x n so, to be more precise few previous samples and the present sample of x n and some previous values of y n ok.

And here we use couple of parameters that a ks and b ls, they are that deciding about the characteristics about this linear equation and, we have a gain factor that G that is giving that how much actually magnification would happen and, that the order of the system is given as p and q. So, look at the p is the number of actually that the terms a ks and that q is the number that numbers of b l ok.

(Refer Slide Time: 01:35)



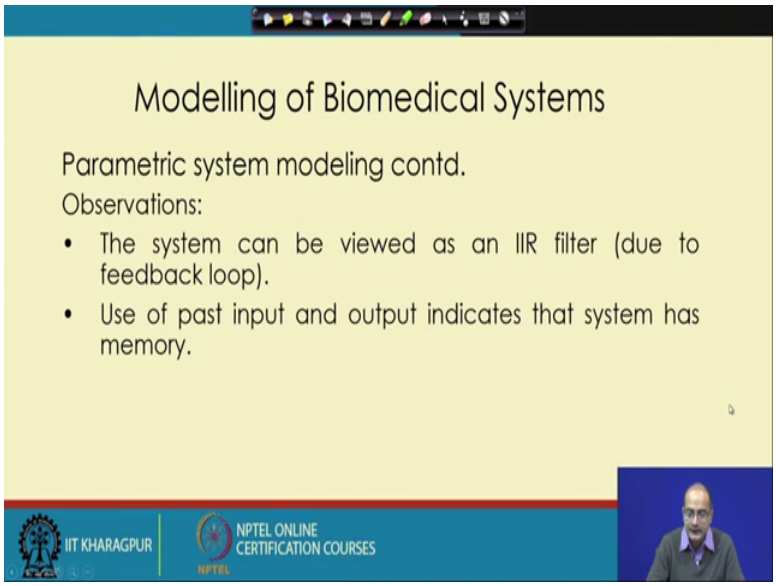
So, that is the way it is defined and what we get that out of these two parts the 2 summation have been given, it is represents the moving average of the input x and that the other part the first summation that is summation over y that is called the autoregressive part, or AR part of the system and altogether we call this system as a auto regressive moving average system or in short ARMA system. Here that x n in this such kind of equation that x n could be any input, but for ARMA model that x n is chosen as the white noise.

The reason is that white noise has something commonality with the impulse that in the previous case in the point process we have seen the impulse the diving that the system. In this case the white noise is chosen in place of that impulse that 1 part of it at is the that in the physical model for example, for Ocado model we have seen that different kind of input sometime the glottal pulse sometimes that the turbulent air coming through from the lung.

So, that turbulent air coming through the lung can be given actually modelled as this white noise. So, those are the physical you can say that significance, but there is a mathematical significance also that in both the case, if we look at the autocorrelation functions, the autocorrelation function is impulse response and, that gives rise to an interesting fact, that when you look at the spectral domain the spectrum is flat. And spectrum is flat means it has all the frequency components in equal measure.

So, white noise is not lacking in terms of any frequency input. So, by now using this equation, we can chose that what part to keep and what part to eliminate ok. So, that that is the way we can say that white noise is a rich in terms of the frequency and it can actually represent help to represent any kind of output.

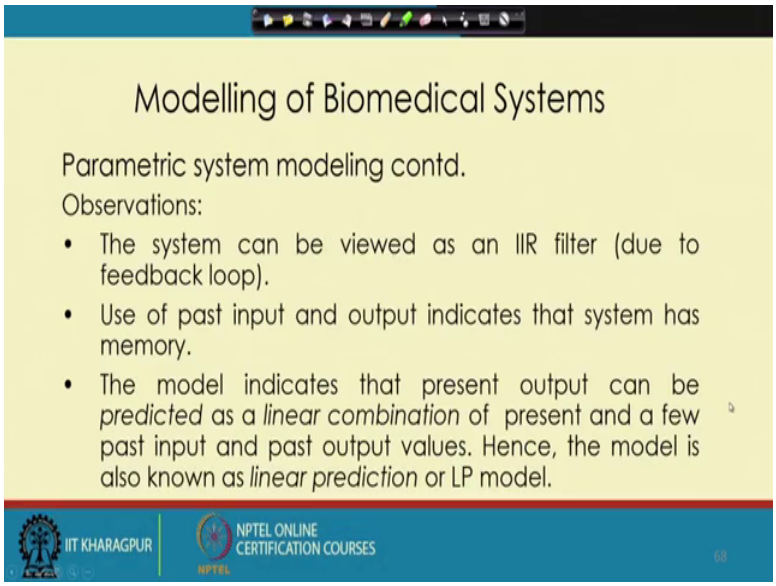
(Refer Slide Time: 04:22)



Now, let us look more into it through observations first we make, the system can be viewed as a in finding impulse response filter, or IIR filter because there is some feedback loop. So, along with that we need to think about the stability and all, but the system is an IIR system first we need to be aware of that.

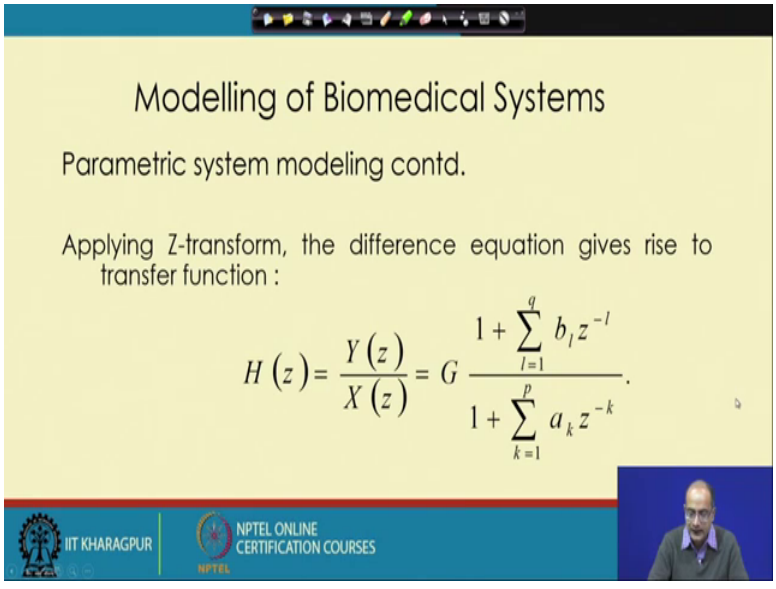
The next point is that the past input and output indicate that system has some memory here, the output depends on present input as well as the past inputs and some past outputs also so; that means, the system has a memory. So, if there is any change in the input that the overall output cannot change immediately because, of that memory it will require some time to get actually those memories are erased, or updated by the new values and then, only we can explain that overall change in the output, or we can say that the change in output will have some inertia.

(Refer Slide Time: 05:48)



Next what we get the model indicates that the output is a linear combination of present and the few past output. So, because it is a linear combination of the the present input past input and past outputs, we can tell this is a linear prediction model or in short LP model.

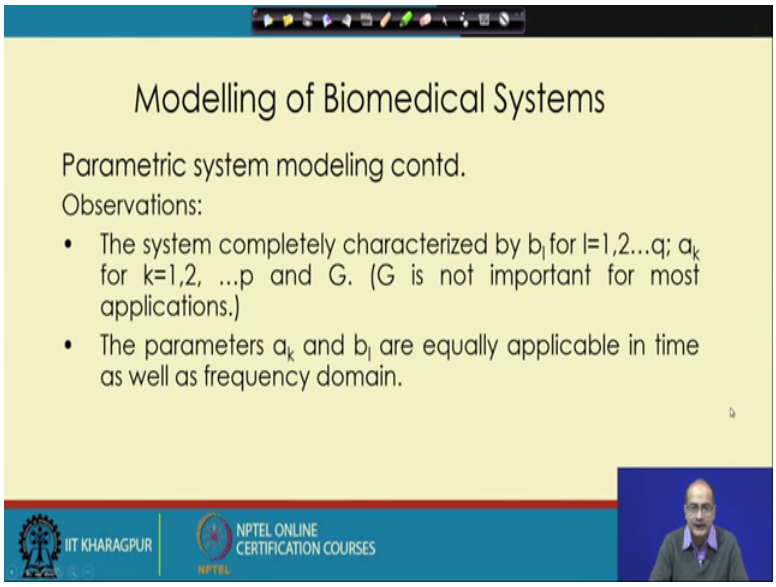
(Refer Slide Time: 06:20)



Now, if we take the z transform these difference equation we get in a new form and, we can derive the transfer function is z out of it and, we get 2 polynomials consisting of a k and b ls that the b l part that we will get in the numerator and the a k parts that we get

that those polynomial in the denominator and we get the gain factor G.

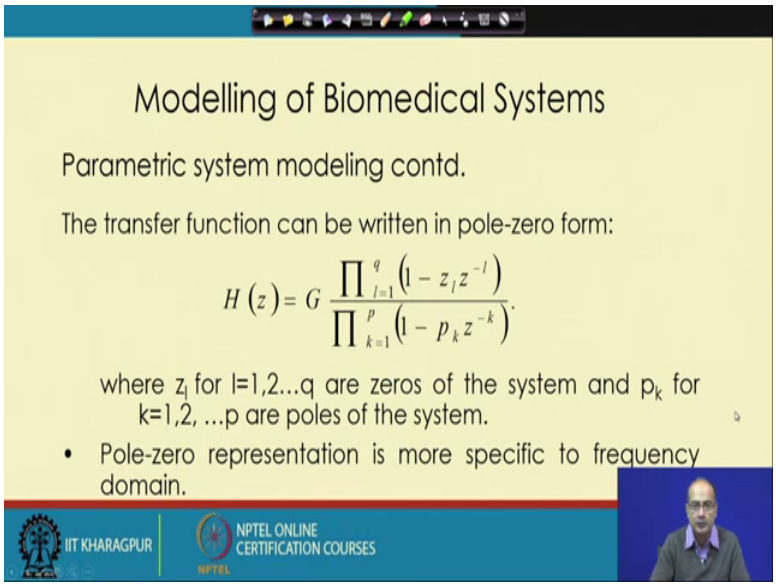
(Refer Slide Time: 06:56)



Now, here again we can make some observation the first thing is the system is completely determined by b l and a case ok, that that using these terms that we can get that the system characteristics G is important, but not so, important because G is gives a uniform scaling it, gives a uniform scaling irrespective of the frequency.

So, if we are not much interested in the exact value; that means, if we get a scaled version of the output that is fine, then we may afford to forget about G. And another interesting thing that a case and b ls, they are equally applicable in the time domain and the frequency domain, in both the case we see them directly and the system can be determined or implemented if we know this value. So, that is another interesting observation.

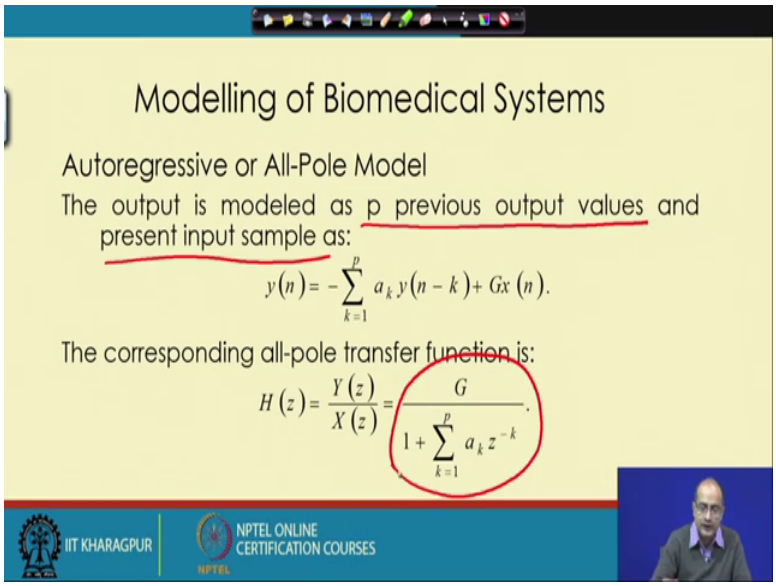
(Refer Slide Time: 08:14)



Now, we can actually represent that transfer function in different form for example, in terms of pole 0 forms. And the same system we can actually represent with the help of the p poles and q 0. Now this is another representation of the same system; however, when we look at this poles and 0s this poles and 0s have equal amount of information as the polynomial coefficients a ks and b ks, but these representation this is most specific to the frequency domain.

Actually if we know the pole location, it could help us to know that where the PHD will have a peak because, that is determined by the location of the poles and 0s will represent that where the that the PHD will take a dip ok. So, we can get more clear information or better indication about the frequency domain using this pole 0 model and this thing does not have any direct counterpart in the time domain ok.

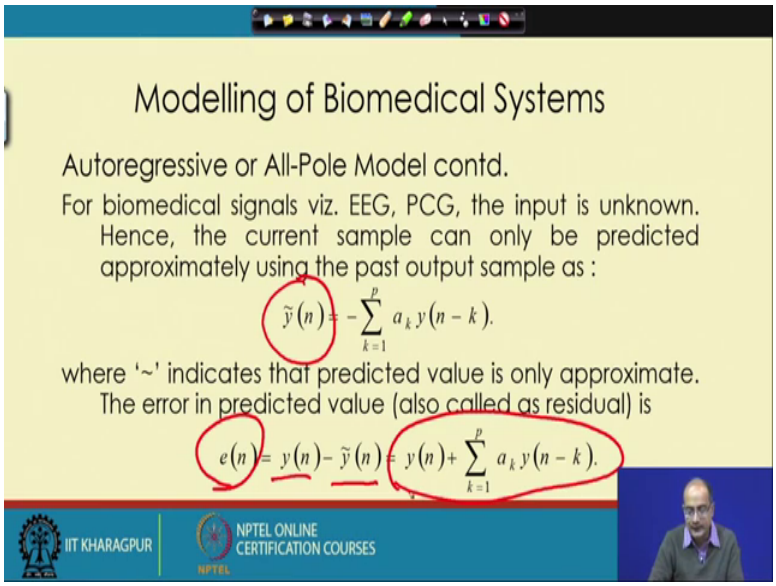
(Refer Slide Time: 09:38)



So, now what we can do we can look at 1 specific part of it, by specific part what we mean that we can first look at the AR part of it or the all pole model part of it, where the output is determined by the p previous output value and the present input sample ok. So, that is the way we are taking only a part of it of ARMA process and we call this that as AR process

So, we are concentrating on that, now here corresponding to that we can have a the transfer function to it, as if you can go to the z domain. And the transfer function here we have represent as transfer function is represented as in this form that, we have that the poles here or in terms of a ks here we have represented.

(Refer Slide Time: 11:22)



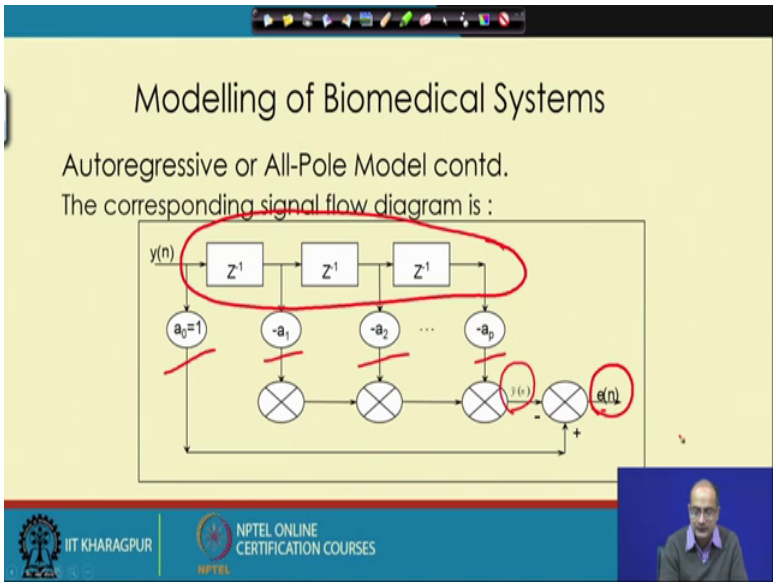
And for the biomedical signal this AR model actually is more common, or more prevalent the primary reason is that, when we talk about the input the input is unknown for the EEG PCG all such cases the inputs are coming from internal body parts which are difficult to capture.

So, input is unknown in that case what we are observing in a non-invasive way the EEG or PCG we only have the output. So, it is easier to model them with AR model ok. So, that is the way we would go ahead with the modelling and for that AR model is the most suitable vehicle.

So, when we try to predict from the past output, we get some predicted value that of y what we have represented as y tilde ok, that the tilde is actually used to tell that it is different from the actual value of y, or it is an estimate of y a and. So, there would always be some amount of error in that prediction and that prediction error is represented by that the term e n which is nothing, but the difference between the true value y n and is predicted value. So, it can be represented by this polynomial ok

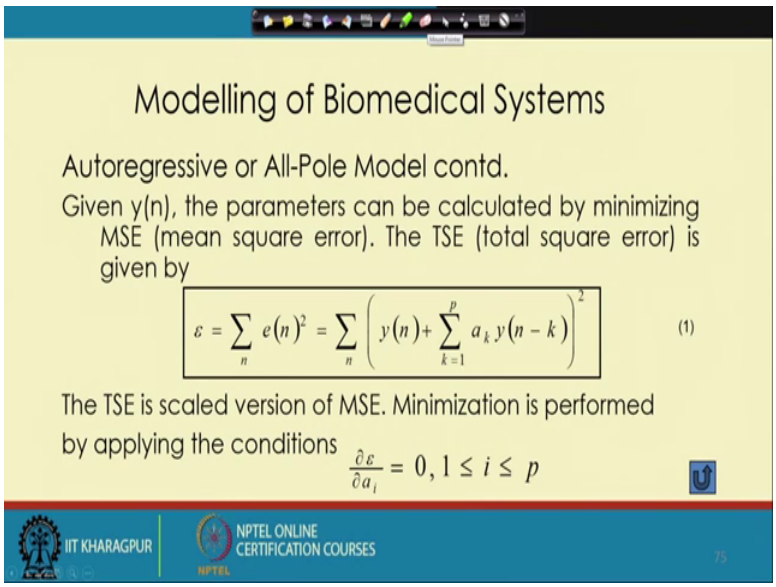
Again if we look at this polynomial the coefficients are a k ok. So, the polynomial is very similar to that what is the polynomial of the AR model.

(Refer Slide Time: 13:47).



Now, here if we have implement it we should look at the signal flow diagram, we get it can be again represented as a tap delay filter, where we have some buffers or registers which would give actually the previous value in a more precise way we can tell that we have a tap delay line, this is the tap delay line which is capturing the previous values of y n and at each step we are multiplying it with some coefficient and, we are accumulating them to get the predicted value here. And if we take the difference of it with respect to y n we get the output e n ok. So, that is the way we can get e n.

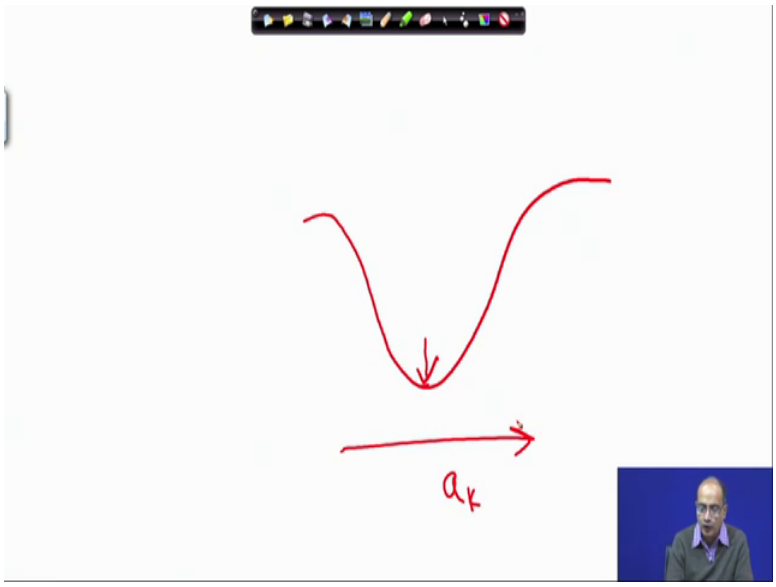
(Refer Slide Time: 14:44)



Now, given y n that is a signal we observe for example, EEG or PCG as you have mentioned in the few slides back. The parameters if we want to calculate; that means, if you want to find out the the system behind it, we need to find out those parameters and that we can do by minimising these total squared error. Already we have defined the error term. So, if you take the total square error; that means, the overall the prediction error for this signal by minimising that we can actually get those parameters.

In other words we want to take, the parameters in such a way that the error can be minimised or the prediction would be very near to the our the true signal. And if you look at that this total square error it is nothing, but it is just a scaled version of the mean square error, in case of mean squared error we have a scaling by the term that is given by the number of samples of e n, that how many predictions we have done we should divide by that. So, by applying that condition so, we can get actually that using the minimum error condition at that point that, we take that value should be 0 here; we should just take out a just 1 minute.

(Refer Slide Time: 16:48)

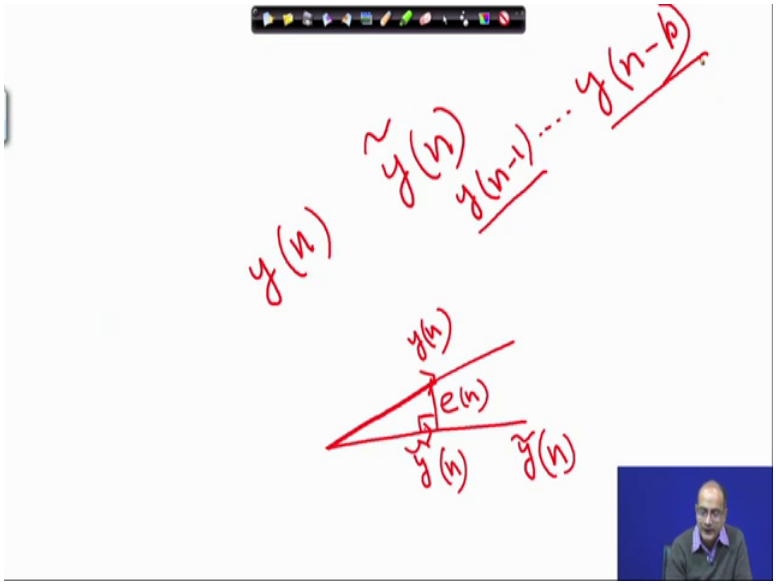


Here the concept is that we are going through a surface and we have some actually minima, there we are trying to find out the minimum and when we reach that minima at this point, then if this is the direction of 1 say a k for that, when we reach that minima at that point the gradient has to be 0. So, we are making use of that property we are taking the partial derivative and we are looking for that minimum point.

So, for the condition is that if we make it the value is 0 that can give us that minima and as we have p such values of a ks or a i s whatever the index we use depending on that that we can get so, many such equations we can get ok. We get p equations and this p equations can be written in this way, in the left hand side we are getting the terms which are having the coefficients a ks in the right hand side that is free from a k. So, p equations p unknowns so, if we know the value that other values which are coming from the signal.

Now, we can compute a ks ok. Now these equation it is known as normal equation that is another interesting point that it is called as normal equation the reason is if we look at that.

(Refer Slide Time: 18:57)

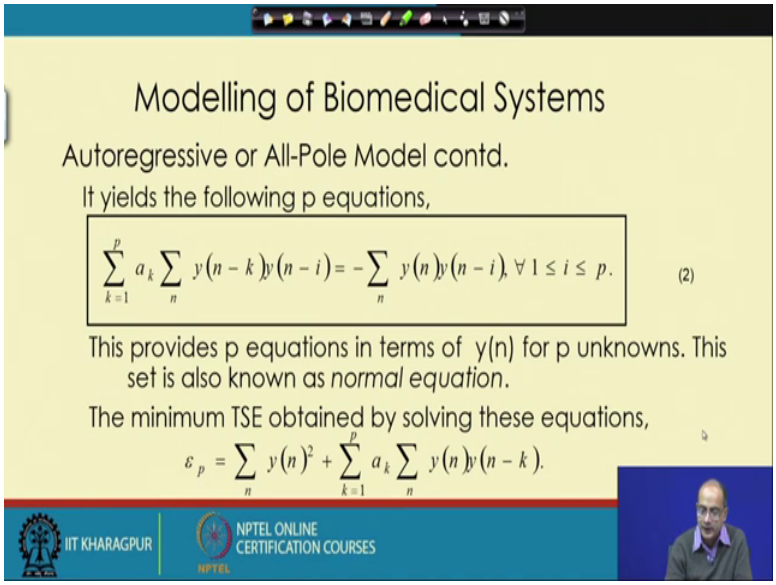


What we are trying to do? We are trying to get actually that y tilde y n and for that we have computed y tilde n and y tilde n is expressed in terms of y n minus 1 to y n minus p. So, it is a linear combination of actually this past values of the output.

So, now we can actually write y as a vector y n say y n is a vector in a particular direction and that prediction that y tilde n is another vector. Now this 1 would be minimum, when we have a actually we draw a perpendicular from y n to these line ok. So, because this is perpendicular and this value is e n this value is e n and this value is y tilde n. So, this perpendicular means when the error is minimum at that time the error e n, it is perpendicular to y tilde n means all the values of y n minus 1 to n minus p, all the

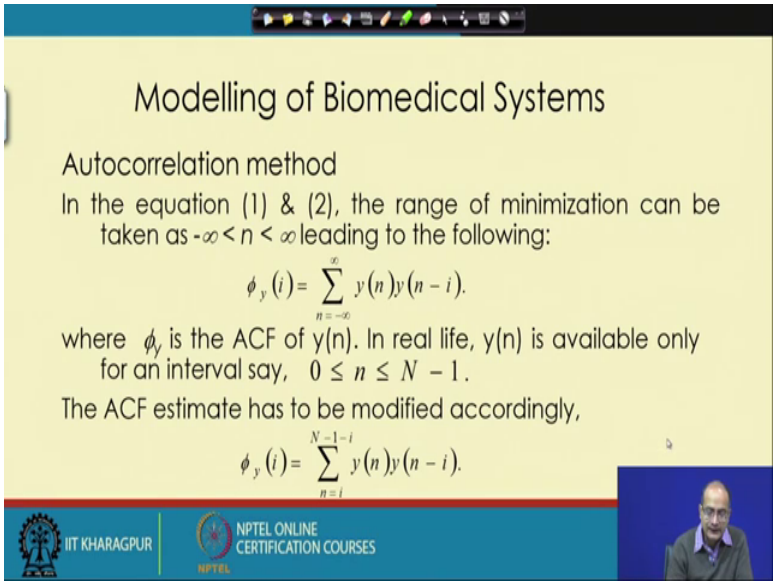
past values it is perpendicular, and from there it came that to know as a normal equation ok. So, that is this the meaning that why this is called a normal equation.

(Refer Slide Time: 21:00)



Now, in this case if we try to next thing would be that what is the amount of error are we have incurred. So, that also we can figure it out once you know the value of the a ks from this equation, if we can find out the a ks y ns are already known. So, we can compute what is the value of that minimum error, which can give us some idea that how the model has been successfully fitted with the signal ok.

(Refer Slide Time: 21:40)

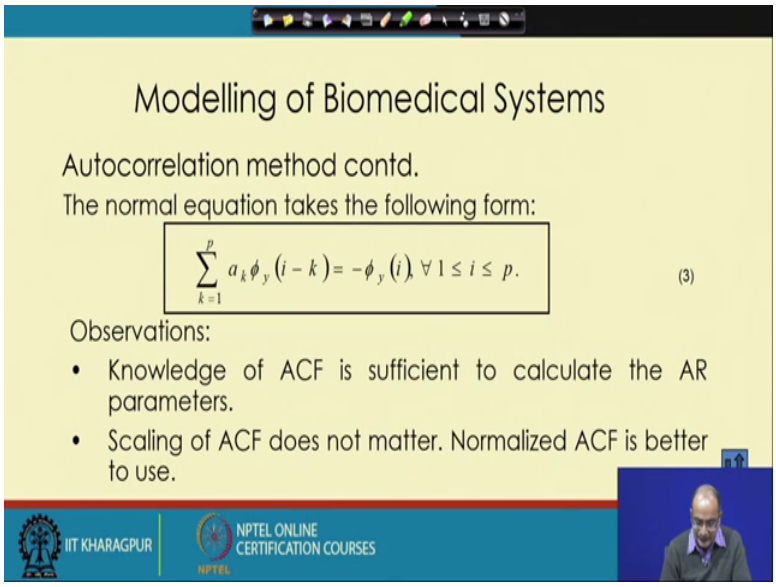


So, given this equation 1 and 2 the equation 1 is the first equation of the AR model and equation 2 is the that, equation required to find out the a ks now the range of minimisation, we can take first that is coming from minus infinity to plus infinity, or you can say the simple intention is we want to actually minimise it over all the errors. And that can give us the form of that the summation is taken from minus infinity to plus infinity. So, when we take the summation of y n and y n minus 1 because, it is a linear it is a shift in variance process, then we can tell it is nothing but the autocorrelation of y at i-th lag ok.

Here we need to again keep in mind that we have assumed the signal a shift invariant, or it is not changing with time. So, that is why we can write it in such compact form and phi y is the ACF. And if you look at what we get in reality in real life, we can only get finite number of samples and to be more precise, if we have n sample we can represented by the value of small n is starting from 0 to capital N minus 1. And in that case we have to modify the ACF estimate ACF estimate would be limited from n equal to i to n minus 1 minus i, the reason is that outside that interval that n equal to 0 to capital N minus 1, we do not value we do not know the value of y n.

So, to keep that summation within that limit, we need to limit the that the interval of summation. So, that is the thing we have precisely done in this case and we have to modify the autocorrelation function estimate what we can get in reality.

(Refer Slide Time: 24:29)

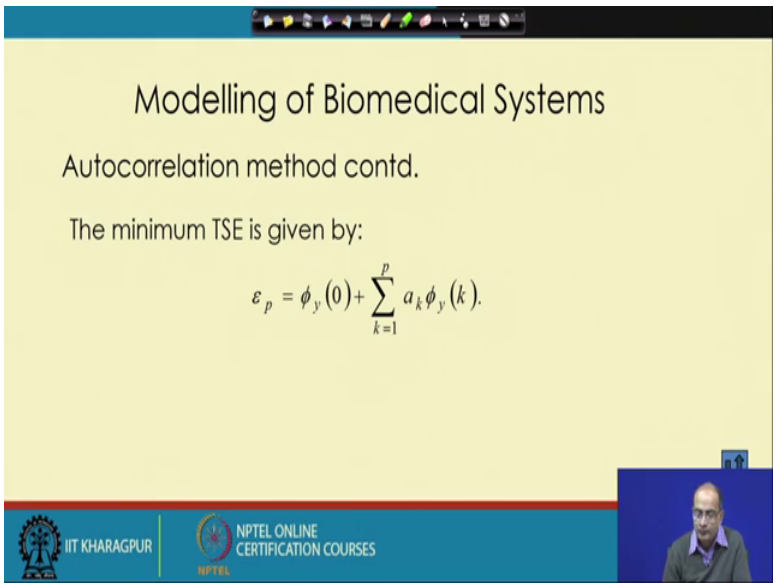


Now, for that we get a little more compact form of the normal equation, we can write equal to 1 to p that a case multiplied by the form that i minus k in the left hand side and right hand side it has again autocorrelation coefficient of that phi that y. So, we have such p equations do we have define and with the help of the p equations, we can actually get the values of a k.

Now, before solving that let us look at some observation that first thing what we note, the autocorrelation function is sufficient to calculate the AR parameters, instead of the signals, if we had only the value of the autocorrelation function of the output signal that is sufficient to find out the model parameters or a ks. So, that is the first thing we note and that is a very interesting observation.

Next point is the scaling of autocorrelation function does not matter because, both the sides we have the autocorrelation function. So, if we scale actually that is scaling gets cancelled; however, the normalisation of the ACF is a better way to compute, those ACFs just to avoid the overflow of the accumulate, we are accumulating the sum of product so, that that does not go beyond the register. So, it is better to actually normalise them from the capacity of the machine point of view, but from the point of view of just say equation any scaling is fine and scaling does not matter.

(Refer Slide Time: 26:56)



And in this case, we can compute the error also in terms of our the autocorrelation function of the output signal and, the values of a ks and the different lags of that phi y.

So, what we get that for the AR model of model order p what all we need to know is p plus 1 lag values lag values of the autocorrelation; that means, we need to know phi y 0 up to phi y p, we need to know that and for a real signal we know that autocorrelation function is symmetry.

So, we get the negative lags also. So, using that we can find out the values of a ks and once we know those values, we can compute the minimum total squared error, also in terms of the autocorrelation function and the a ks. So, this is the brief module on the that autocorrelation function.

Thank you.