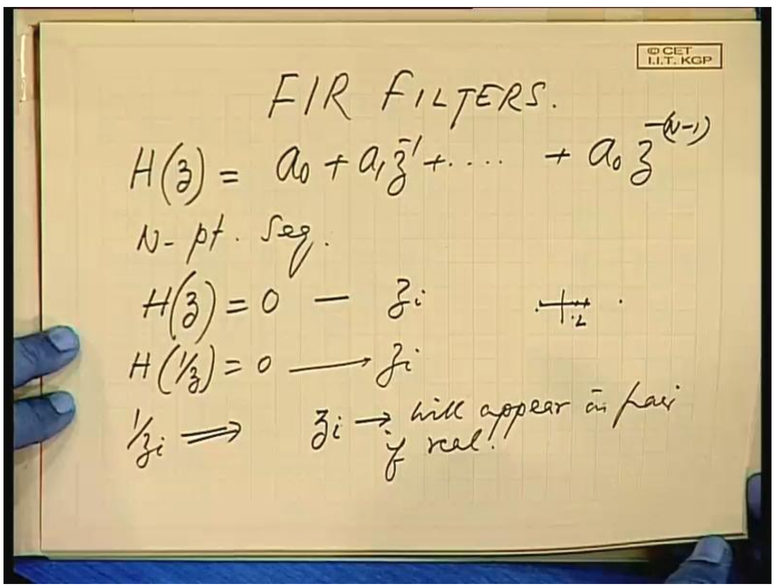
Digital Signal Processing
Prof. T.K. Basu
Department of Electrical Engineering
Indian Institute of Technology, Kharagpur

Lecture - 15
FIR Filters

Today, we shall be taking up FIR filters.

(Refer Slide Time: 00:48)



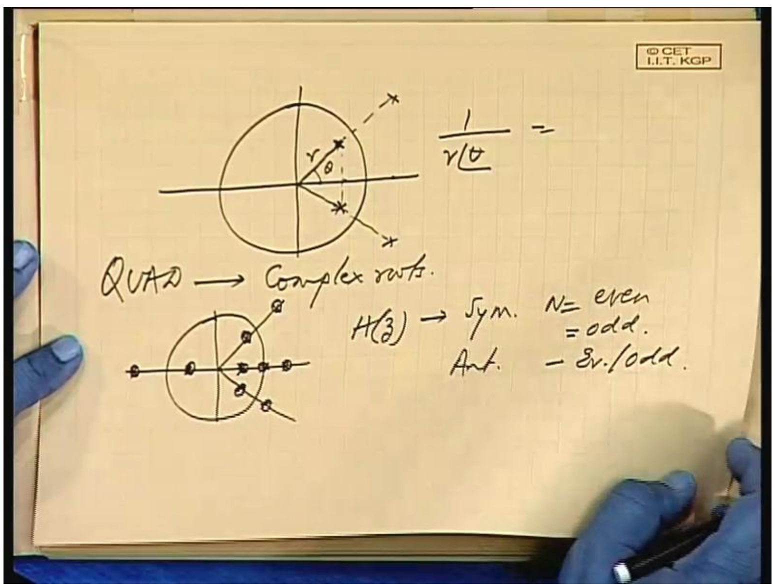
As we discussed last time, let me first of all revise in short what we did last time. H z for linear phase characteristics, we had symmetry of distribution of the coefficients; say last one is a 0 z to the power minus N minus 1, that is a N point sequence. We we studied different situations. If there is symmetry and if N is even or odd, what will be the expressions?

And then if it was an anti-symmetric sequence; that is if this is a 0, this is minus a 0 just opposite signs from the other end, if you find with equal magnitudes of the coefficients then that is an anti-symmetric sequence, we also derived the conditions for that. Now, there are certain peculiar properties of this.

Let us study them in brief then we go over to the design of FIR filter. H z is equal to 0, gives me the zeros of the polynomial, all right. You see, H 1 by z is also 0 at those points or in other words; 1 by z i is also a solution that means, if z i is a root then 1 by z i is also another root, except at 1 or minus 1 because inverse of 1 is 1, inverse of minus 1 is also 1 minus 1.

If it is having a root at 0.2, then it will have a root at one by point two that is five also. So, z z i's I except at plus 1 and minus 1, we will have pairs of roots, z i will appear in pairs if real, okay. Suppose the roots are complex, then let us see; this is just a unit circle, there is a complex root r theta.

(Refer Slide Time: 03:34)



Then r minus theta is also root okay. Since this is a root 1 by r theta, this is also another root; so 1 by r theta, theta will become minus theta, so is a root here and 1 by r minus theta will be here. So, basically there will be two pairs or a quad for a complex roots, okay.

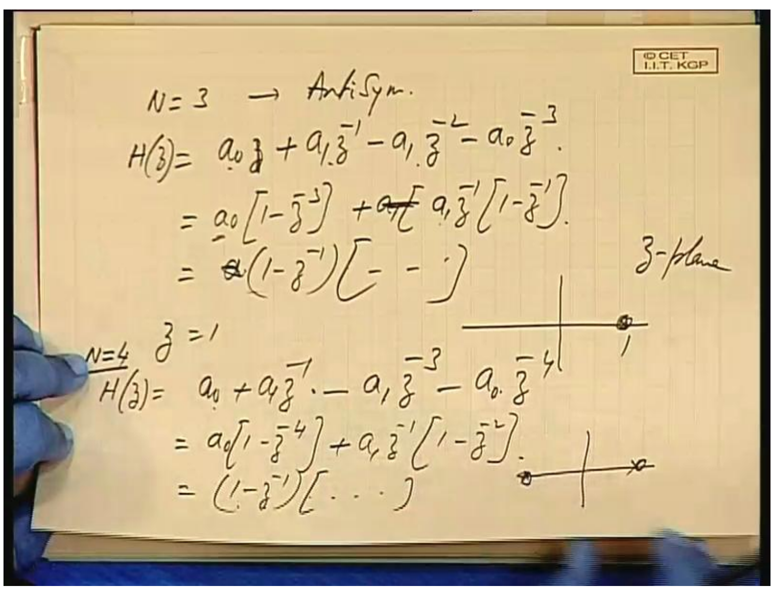
For complex roots there will be, four such roots coming in a group. For real roots, they will be in pairs; so the distribution of roots will be like this, depending on the size of the polynomial. There can be say 1 root on the real axis, then there can be a few pairs on the real axis other than one.

And there can be groups of four's like this, I should have put zeros to indicate them as zeros; instead of showing them just a star as roots, there all zeros of the function.

So, it will be better to use zeros that is a standard symbol, okay. So zeros distribution will be like this. Now, you can try out what are the different possible locations of roots; what are the possible locations of roots for H z, symmetric; all those four guesses and N even or odd then anti-symmetric, again N even and odd. So there are four cases for which you try out what are the possible distributions.

Now for an anti-symmetric case, there is a clear cut case. Let us take an example of say; N is equal to say 3, N is equal to 3 means, it will be a four point sequence. So and we are considering anti-symmetric, anti-symmetric sequences.

(Refer Slide Time: 06:26)



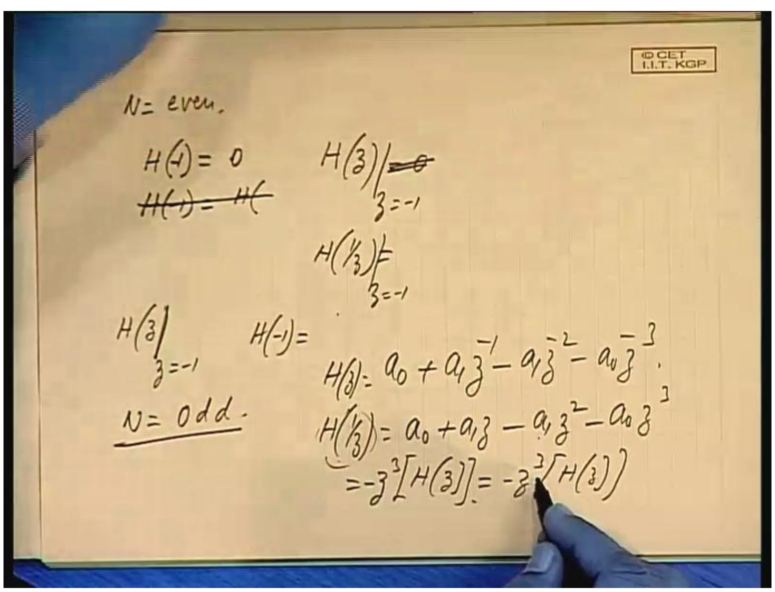
So, H z will be a 0 plus a 1 z inverse then minus a 1; so 3 4 1 2 3 4, z to the power minus 2 minus a 0 z to the power minus 3, correct me if I am wrong, this is all right. So, I can take a 0 into 1 minus z to the power minus 3, is it all right; plus a 1 okay a 1 z inverse I can write, a 1 z inverse into 1 minus z to the power minus 1, is that all right.

So, 1 minus z to the power minus 1 is a common factor, other factors will be depending on the magnitudes of a 1 and a 0, 1 minus z 1 is a common factor. So, z is equal to 1 is a root all right; that means there will be a root here at 1, this is the z plane. Let us take N is equal to 4; this is irrespective of the coefficients a 0, a 1, if I have H z is equal to 4, N is equal to 4 then H z will be a 0 plus a 1 z inverse then there won't be anything then minus a 1 z 3 and a 0 z 4, okay.

So, the number of terms though they are four, but actually it is a five point sequence because it runs up to z to the power minus 4, there is a zero coefficient in between, okay. So, here it will be a 0 into 1 minus z to the power minus 4 plus a 1 z inverse into 1 minus z to the power minus 2, okay. So, once again 1 minus z to the power minus 1 is a common factor.

There can be other factors. In this case it is 1 minus z to the power minus 2, all right. So for N is equal to 4, for N is equal to 4 it will be 1 minus z inverse into something; in this particular case, it will be also 1 minus z 1 plus z inverse, so there will be a root here, there will be also root here, a zero here, okay. So whether it is N is equal to 3 or 4 you can go to, N is equal to five, six. You will find this root is common, is that all right? What about H is equal to 1 for N is equal to even?

(Refer Slide Time: 10:10)



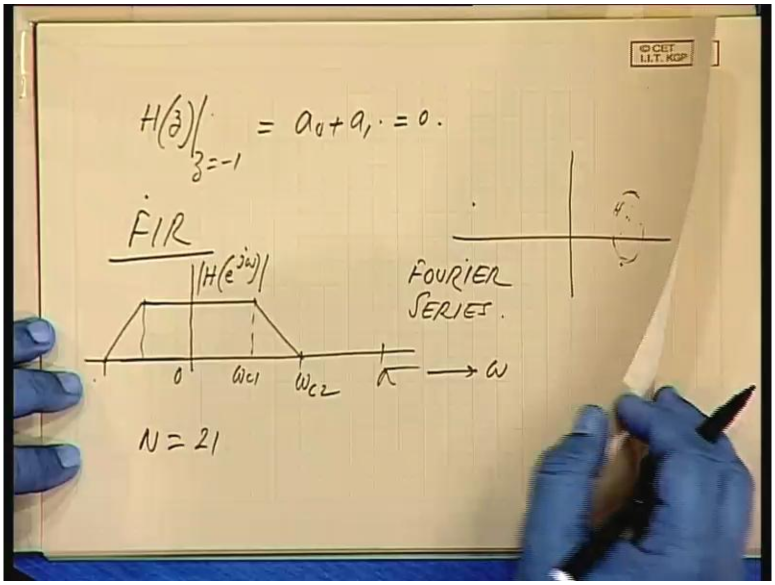
What is the value of H equal to 1, H equal to minus 1? If I put yes please, H is equal to minus 1; it will be always a 0 minus okay a 0 minus a 1. For N even H minus 1 will be always 0, can you see that? H minus 1 will become equal to H plus 1, just see okay all right; no, not necessary, not necessarily, sorry I withdraw.

Say, H z is equal to 0, what is H z? At z is equal to minus 1, what is H z at z is equal to minus 1? Middle term is missing. So, it will be a 0 plus a 1 into minus 1 minus a 1 into minus 1 a 0 into plus 1. And what is H 1 by z at z is equal to minus 1? I compute that, H of z at z equal to minus 1. So that means, H minus 1 is equal to; H with the negative sign, for N is equal to odd I am sorry, for N is equal to odd, I am sorry for N is equal to odd H z, no but we are not getting that, well I think we will have to check N at 1 plus z, okay.

N is equal to odd, for N is equal to odd we have something like this; a 0 plus a 1 z inverse minus a 1 z to the power minus 2 minus a 0 z to the power minus 3, is it all right okay. Now, how much is the value of this function? This is H z and what is H of 1 by z? a 0 plus a 1 z minus a 1 z square minus a 0, z cube.

If I take z cube outside, minus z cube into this will become a 0 plus a 1 z inverse minus a 1 z 2 a to the power minus 2 that is H z, is that all right. So, it is minus z cube into H z. I evaluate this at minus 1, it will be minus 1, this will be just minus 1.

(Refer Slide Time: 14:09)



Say, H z at z is equal to minus 1 is, how much? a 0 plus a 1, if I put z is equal to minus 1, a 0 plus a 1 minus a 1 this one; a 0 plus a 1 minus a 1 minus a 0, how much is it? 0, okay. So, for this 1 minus z inverse was a factor, I did not evaluate this but I find at minus 1 also it is vanishing. So, there will be a 0 here also, know.

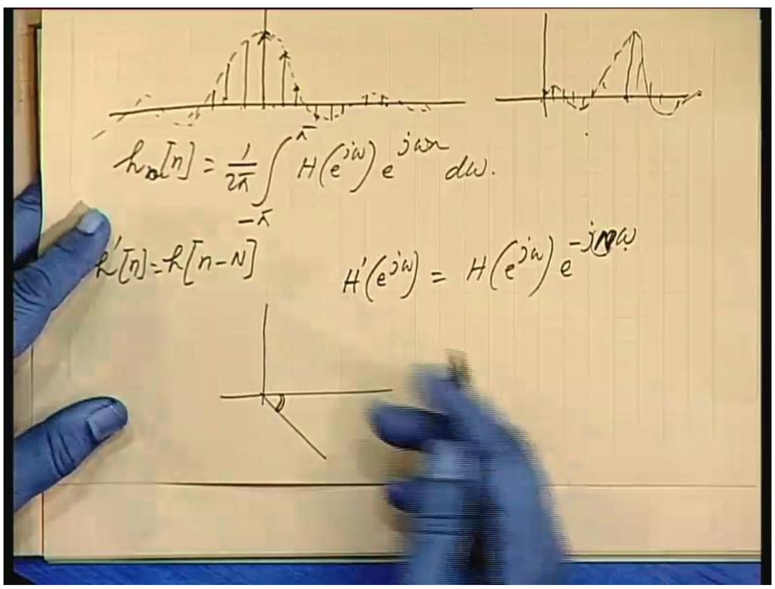
Just now, you have seen for N is equal to odd N is equal to odd, is it not. If I put z is equal to minus 1; this is vanishing, these terms will get cancelled, sorry N is equal to, N equal to even this is one, for N is equal to even, it will be cancelling. For N is equal to odd, it will not be sorry, does it cancel? May or may not.

So like this, you keep on taking different polynomials and then see the distribution of possible poles and zeros. Now, yesterday we were discussing; if you have a zero or a pole or pole zero pair, very close to the imaginary axis to the sorry, very close to the first quadrant and fourth quadrant or third quadrant and second quadrant, there are there are characteristics similar to your band pass, high pass, low pass and so on.

So, if you are given a particular type of filter, you know the approximately what will be the location of zeros; if it is an all zero function and then you can form, whether it is an, it is going to be symmetric or anti-symmetric sequence. Now, let us come to to FIR filter design. What we discussed last time is; we just gave an introduction to Fourier series approach.

If you are given a function say like, this this is the given characteristic; that is the desired one some omega c 2, this may be omega c 1 and this is p i and similarly, these two minus omega c 1 minus omega c 2. Then you are asked to design a filter of, say N is equal to 21 okay, total length 21. As I was mentioning for any function, this is a periodic function, is it not? It is a periodic function, with the period of 2 p i.

(Refer Slide Time: 17:53)



So, I can always get the Fourier coefficients h n, as 1 by 2 p i minus p i to plus p i, H e to the power j omega e to the power j omega n d omega, okay. Now, this sequence this sequence that will be coming here; say may be like this over a profile like this, if you take a discrete version of this okay. It is a sink function, in case it is a rectangular function. H omega is a rectangular function then it will be a sink function; it will slightly different, if you have a roll off like this, all right, so it will be like this.

When you are given N is equal to twenty-one, so I will have twenty-one coefficients chosen from here; ten on this side, ten on this side, one at the centre so that will become twenty-one. If it is a twenty point sequence is a slight difficulty. Normally, we take an odd point sequence, I will tell you why. Anyway, you can take any even point sequence but then this sequence you have to choose will have, to be chosen a dropping this.

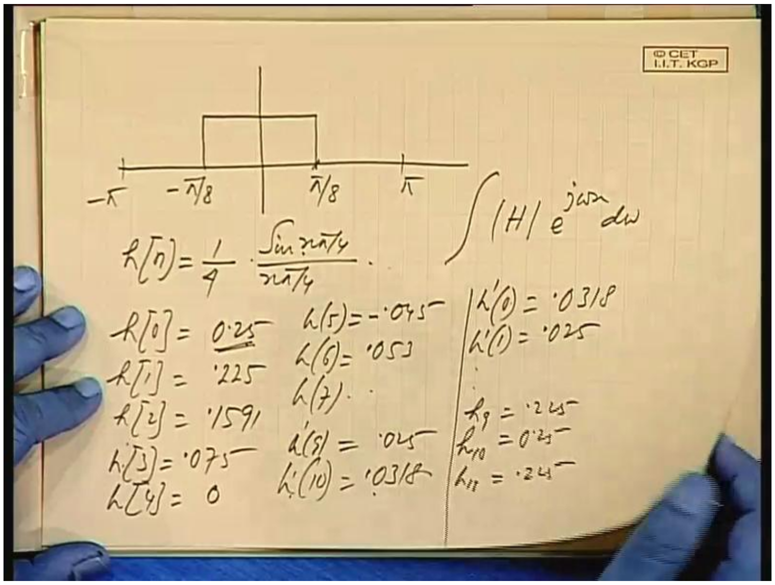
Now, if I give a shift of ten steps what happens? If I give a shift to any sequence h n; this h n that I have got is non-causal, it extends on the left hand side also. Now, if I give a shift of ten steps then I will get a causal sequence and the sequence looks like this, it may be, points coming out of this.

This goes to the eleventh position all right. So, there are ten values; below this ten values, above this okay. So, I get a symmetric sequence like this. So, if I give a shift towards positive side by ten steps, what is the net contribution of the function, phase? It will be, suppose I call it h dash ten, then what is h dash ten; is h n minus capital N.

So, H dashed e to the power j omega; will be nothing but original e to the power H e to the power j omega, e to the power minus j 10 omega, so this is n omega okay; n is 10 but generally it is capital N, is that all right. So that gives me a linear phase that is given here, okay. So, if you are given a linear phase characteristics then you know; this one has to be some N omega, so if the slope is given, say fifteen then you know you can choose a filter of fifteen plus fifteen plus one, thirty-one, length of thirty-one, okay.

So, you take the Fourier coefficients and then give a shift. Let us take an example, it will be clear. So, we shall now consider a function.

(Refer Slide Time: 21:48)



This is p i by 8, minus p i by 8, this is p i minus p i, okay. So, you compute h n h n is suppose you do not give any shift to start with; you just integrate H magnitude e to the power j omega n, as if its phase is zero. Whatever coefficients you get, is shift it by so many steps, you get the desired filter.

So, 1 by 4 sin n p i by 4 by n p i by 4, okay. So, put different values of N and compute h. Say, N is equal to 0, so h 0; N is equal to 0 except that, N is equal to 0 other values will be very easily computed. So, this is 0.25, N is equal to 0, sin 0 by 0. Basically, sin theta by theta, theta tending to 0 so that is the limit, 1.

h 1, h 2 and so on, you compute. h 1 will be sin p i by 4 by p i 4 into 4, so that gives me 0.2250 okay. Sin p i by 4; p i by 4 into 4, so sin p i by 4 by p i, so that is about 0.225. Then this is 0.1591 and so on. I have got some values listed here; h 3, 0.075 check whether these values are all right. When you come to 4, N is equal to 4 sin p i that is 0.

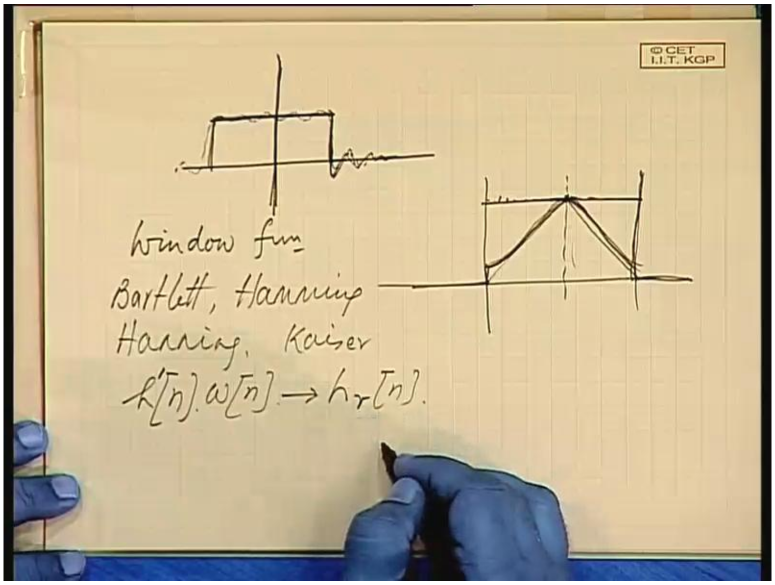
So, occasionally as you have seen earlier, some of the values will be touching the zeros; so there will be if they are multiples p i by 4, multiplied by by an integer gives me a value like sin p i, 2 p i, 3 p i and so on, it will become 0. So, periodically we will find zeros are appearing, h 5 and so on, minus 0.045, h 6 similarly 0.53, h 7 anyway you can compute ten values. h 10 comes to 0.0318, okay.

Now, after you have computed this, you give a shift by ten steps then what you get? This will shift to the central point all right, as we have shown the peak has shifted to eleventh position here. So, this will be the last point. So, it will start with 0.0318. So, I will call that as h dashed 0 as 0.0318, okay which is nothing but old h 10.

h dashed 1 will be 0.025, just before this h 9 was 0.025 and so on, is that all right. h 10 will be 0.25, h 11, point we started with 0.25, is it 0.225? 0.225. No, 0.025, 0.25 is a maximum value; so h 10 will be that, okay. It starts with h 0, so h 11 will be 0.225, is it all right? h 9 is also 0.225, it is all symmetric. So, this is what you get as a filter.

Now, what you have done? An infinite sequence has been truncated at tenth step; otherwise it stretches up to infinity, plus infinity and minus infinity. We have truncated it at that level; say ten twelve anything you want. What happens to the final response, does it remain same? It does not. So, the response that will be, that will be getting from these ten truncated values; rather twenty-one point sequence, will not match with the actual value.

(Refer Slide Time: 27:14)



So, actual desired function was like this. Had I had those components which we have dropped; then I would got this perfect function like this, since they have been dropped that means, those frequencies are missing. So, the realised filter will have the characteristic like this. So, you will find the high frequency terms present here all right. The high frequency terms present here because you are dropping them, they are conspicuous by their absence.

In this class, there are so many absentees; so I can make out that mister such and such who was sitting here everyday is absent, so by his absence, he is making his presence. I mean, he is reminding me of his face all right. So, it is like this, it leaves an impression that this is absent. So, overall function will become like this, you have to reduce this.

So, what we are truncating was a very abrupt truncation, suddenly at ten steps you are saying stop, no more beyond this. Instead of that, you gradually reduce the importance of higher order terms and then make it zero, all right. So for this purpose, sorry to smoothen this to smoothen this otherwise at sharp changes, these ripples will be very high.

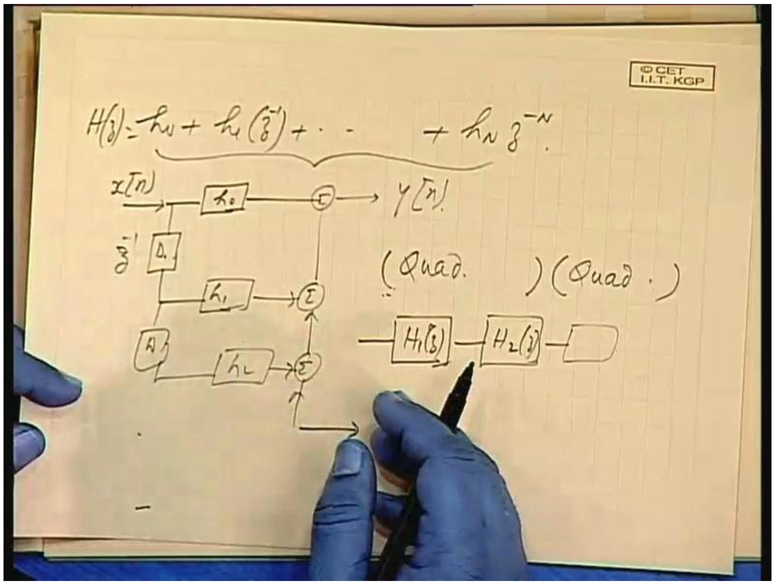
So to reduce this, we use a window function. That means, you have a function; suppose this is a mid-point, we give maximum weightage to the midpoint, value of one and minimum weightage to two ends all right. Somebody may go by a triangular function like this. The one that we have chosen earlier is a rectangular function; that means we have given equal importance to all of them, suddenly were made a zero value at this, okay.

So, this is known as a rectangular window, the one that we have considered so far, rectangular window. Then there are many window functions; Bartlett, Hamming, Hanning, Kaiser and so on there are good number of windows, standard window functions used, more or less they give a similar kind of results.

Now, some of them may be like this. I need not start from 0, I give with a little weightage and then it is like a something like a parabola. So, you will find the distinction is the difference is very very small between different windows. So, h n if I multiplied by w n okay will be realised values of n; that I should be using. I am using h r and this is h dashed, because that is the sequence that we got after shifting it.

So, h dash 10 is multiplied by a corresponding window function, we generate the realized filter functions, is it all right. Now, how do you realise it in the; say how do you write a software or even in a hardware? You have got h 0 plus h 1 z inverse; let us once again come back to a general filter function like this, h n z to the power sorry, z to the power minus 1, z to the power minus N.

(Refer Slide Time: 31:59)

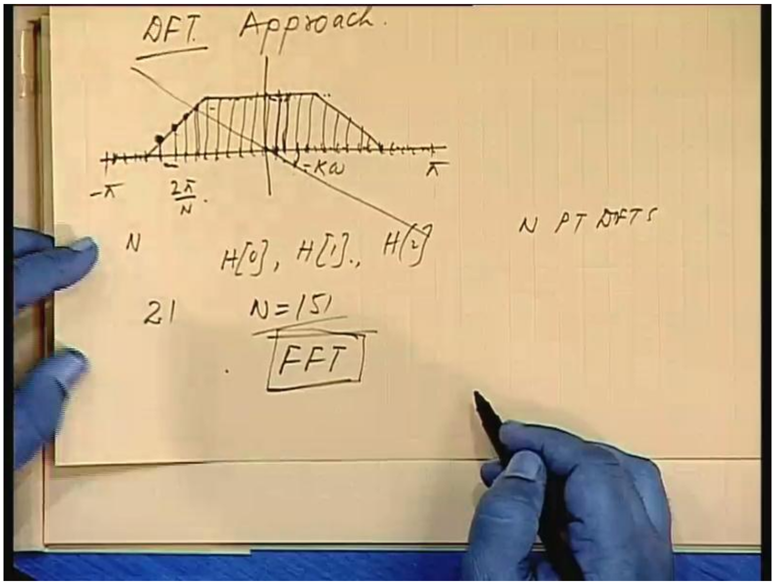


I can give an input x n all right, multiplied by h 0, just a constant. Then I put a delay circuit. Delayed unit, I can write capital D or z inverse as I told you, both the convolutions used by text books and then again multiply by h 1, again put a delay multiply it by h 2 and so on. And then from the bottom, you keep on adding is an adder, again this is an adder, again this is an adder and this will be the output, this is simple FIR structure.

One may be interested in breaking it up into factors. Suppose, I put it in the form of quadratics all right. I factorise in the form of quadratics, if it is possible or may be a bi-quad, sorry in the form of a polynomial of four coefficients; that is consisting of the quads of points, after determining the zeros I can always take polynomials of that order. Then I can have one block H 1 z, may be consisting of four roots, another block consisting of another 4 z, so H 2 and so on put them in cascade.

So, there will be 4 elements here, one two three four, four delay elements here then that will be forming one block. Again put a similar block, so this y 1 will go to the quad input to the next block and so on, so at the end you get the final output y n. One may break it up into smaller blocks like this. There can be different arrangements of this, one may start from the other end and see the output here. These structures you try yourself, what are the other possible structures, okay. Now, next we take up DFT approach.

(Refer Slide Time: 35:13)



So, the Fourier series approach was very straight forward, you take the Fourier functions; Fourier coefficients, truncate it at the desired base and then shifted by so many steps. In the DFT approach we do not consider, basically what you are doing is almost identical to Fourier series approach; there you have integrated as you evaluate the Fourier coefficients in the normal time function. So, over one period of 1 by 2 p i say I mean period of 2 p i.

Here you take discrete values of the function, once again over the period of 2 p i. So, suppose you are having a filter function given like this, desired function like this, this is minus p i, this is plus p i. So, I can always have large number of points say N, large number of points at intervals of 2 p i by N, like this. So, these intervals will be 2 p i by N. So, there will be many zeros okay.

And I am taking the values only at these points, is it not? You are given a particular slope all right; stretch it on the side okay, so this slope is known. Then you if you know the slope, you can associate the angle say; K omega and you can evaluate the value of the angle at every point, corresponding values. So, what I am giving you is H at 0, it is a DC value. H at 1 which is this magnitude and this much angle. H 2 similarly, this magnitude magnitude; in this case remains constant for a substantial length and then this much angle.

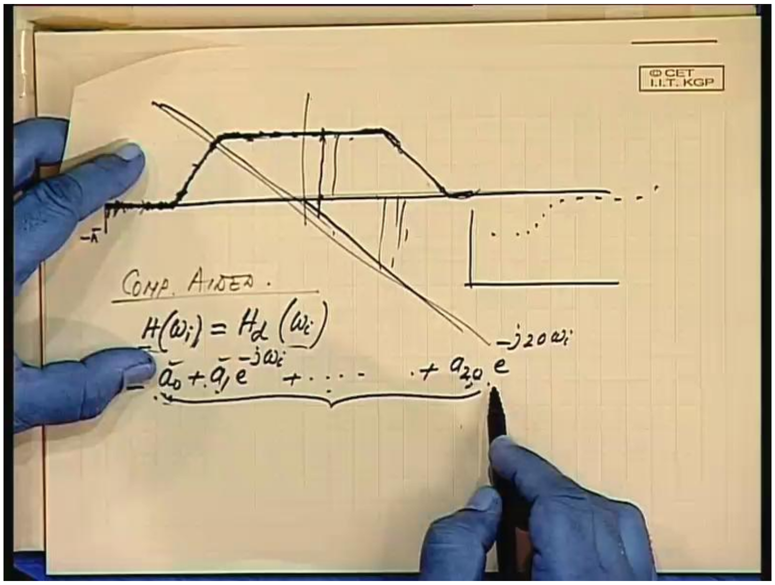
So, like that up to this point you can come; after that the magnitude becomes zeros, so even if it has a phase you do not have to bother. Now, if these points are given then you compute, the N point DFTs, could be the DFTs of this there will be an N point sequence all right. So, if you are given 21 points, twenty-one points sequence then calculate, what will be the angular shift; I have taken those twenty-one points to match with the desired characteristics.

But we do not know what happens in-between; it can be anything all right. So, this will be much more meaningful; if you have a very large number of points, if you take a filter of a very high order then this approach will be very effective because, I will make it pass through a very close points, closely located at a regular intervals. So even if there is a deviation, the deviation will not be much because I am trying to hold it at very regular intervals, all right.

It will be perfectly matching at those points; unlike the Fourier series approach where we are not knowing because suddenly truncating, depending on the type of window, there will be ripples. Here there will be ripples, but there will also confirm conformation at regular points and those points can be many. So for a for a very high order filter, say N is equal to 151 if I take; so you can calculate the filter sequence.

Now for this, we normally go for a very large size of large value of N. Computation of DFT we resort to, FFT algorithm, FFT algorithm. In the next class, we shall be taking up FFT computation all right. So, today we will be discussing about the other two methods. The third one is computer aided design.

(Refer Slide Time: 40:43)



There can be different approaches to computer aided design, let us see one of them. I have a transition here, you see there is a sharp change here, there is a sharp change. Here is a sharp change, here and here. Here it is regular; it is kept at a constant level. Here it is changing at a regular level all right. So, I need not bother about these points so much, as much as this is one, this is of more concern.

So, I take more number of points here; that means I do not take points at regular intervals all right. Again more number of points here, fewer points here; more number of points here, more number of points here, okay. Say let them be starting from minus p i omega 1, omega 2 and here also I need not take so many points, it remains flat, so I may take two three points here, okay.

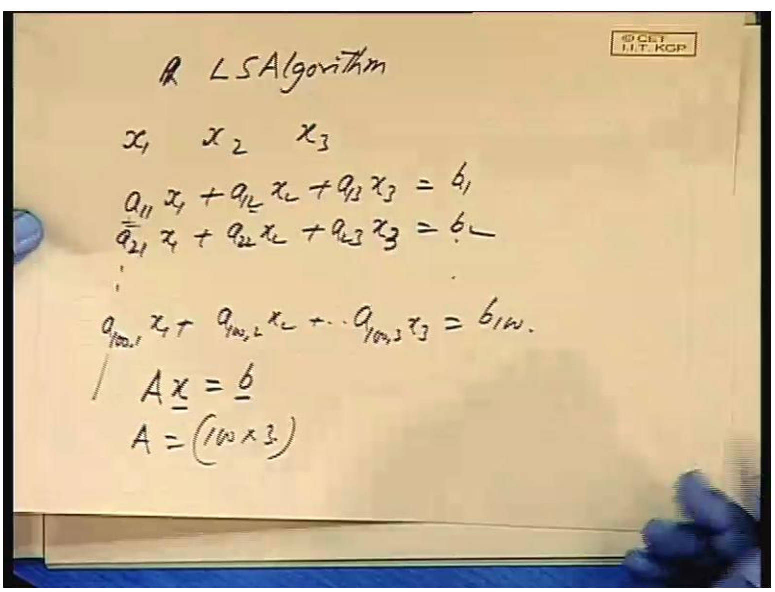
So may be ten points here, ten points there, one or two points here, a few points here again ten points here, ten points here and so on. So, H omega 1, omega 2, omega 3; that I choose that is h omega i, should be matched with the desired value, the ideal one. This is the H realized filter. Now, suppose I take this as fixed number of coefficients; we can have fixed number of coefficients, say I want twenty-one coefficient filter, so a 0 plus a 1, a to the power minus j omega i, okay.

Similarly up to a 20, twenty-one point sequence means, it will be a twenty; e to the power minus j 20 omega i, omega i is this setup points may be hundred points. So, we are determining twenty-one coefficients in a least square sense, least square error sense, okay. Obviously with 20 coefficients, if you want to match exactly at those points; then I need for hundred points, I need hundred coefficients is it not? Unless, I take a up to a hundred all right, a ninety-nine, I cannot really match at hundred points.

So, what I am trying to resort is; resort to is a given a large number of points is something like this, I am trying to fit in the best possible straight line or best possible parabola where the number of coefficients will be two or three, whereas number of points are many all right. So, I am trying to fit in a particular order filter, all right.

This is of twenty-one coefficient and the number of points taken is may be hundred or two hundred okay. So, how do you go about it? What what would be these parameters values? So here we will take up, we will digress a little from here, we will take up what is known as recursive, okay a Least Square Algorithm.

(Refer Slide Time: 44:56)



Sorry mind you, here we will be given, because it is a twenty-one point sequence; basically the slope is ten omega, I want also linear phase characteristics. These coefficients though they are appearing to be twenty, twenty-one but basically they are eleven coefficients because this a 0 has to be a 0 here, okay.

So, there are basically eleven coefficients. So, you might as well put a 0 here, a 0 here, a 1 and a 1 here and so on. Now for each point given, H omega 1 is a complex number that means; at this frequency say omega, omega fifty-one, this is the magnitude and this is the angle, fifty-two this is the magnitude, this is the angle. And I have taken larger number of points here, so correspondingly you have to calculate also the angle.

And then only this, so this is the basically complex equation all right, with real coefficients. Suppose you have let us take a very simple example. You have a, this a very common example, that I give for a discussion on this. Suppose, in this room we have got three heaters all right, at three different locations, three corners of this room and you want to measure the temperature of those three heaters.

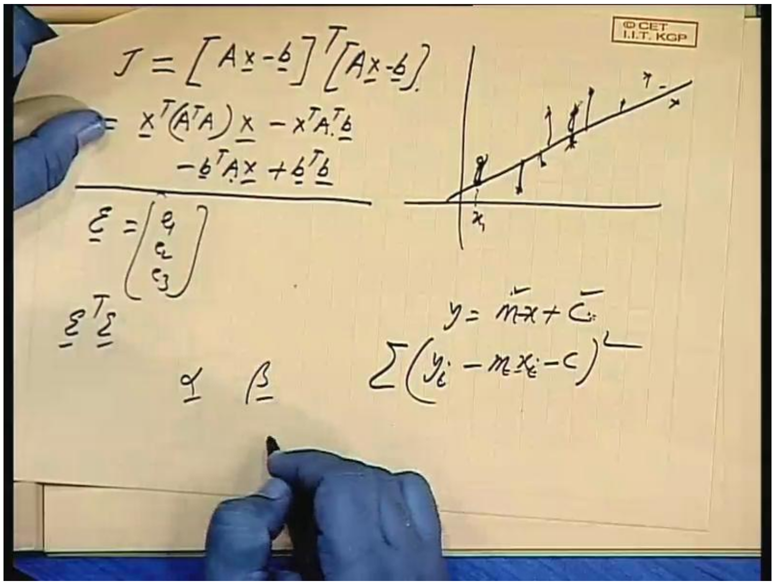
Let them be x 1, x 2, and x 3 okay. And the temperature at any point is a linear combination of those three temperatures, x 1, x 2, x 3, okay. So, at different locations you keep on making measurements of temperature. So at this point, it will be some constant into x 1, some other constant into x 2, some third constant into x 3 and that will be the noted measured temperature, as something says t 1. I will call it a 1 1 x 1 plus a 1 2 x 2 plus a 1 3 x 3 is equal to some measurement b 1, all right.

Assuming that the temperature, here we can apply a superstition theorem x 1, x 2, x 3 and the dependence is linear. It may be inversely proportional to the distance. So a1 1 is known a 1 2 is known, depending on the distance I know this value at different locations, I make keep on making the measurements. So, a 2 1 x plus 1 a 2 2 x 2 plus a 2 3 x 3 is equal to b 2. If my measurement is perfect; if my measurement is perfect then probably I will require only three, any three equations to solve x 1 x 2 x 3.

If there is a very noisy environment, if the measurement is not perfect; there can be error due to my observation, there can be error in the thermometer, there can be some wind blowing here and there. So there can be various sources of errors. So, if we assume the error to be say Gaussian, white Gaussian noise; it is a very noisy and random noise then b 1, b 2, b 3 these are all noisy measurements, corrupted.

I make say hundred such measurements, a 100 1 x 1 plus a 100 2 x 2 and so on a 100 3 x 3 is equal to b 100. I made hundreds of such measurements all right, what will be the best possible estimator the temperature x 1, x 2, x 3, okay? So, you write this as matrix A, vector x is equal to vector b; this is a measurement vector, x is the vector of the unknowns and A is a matrix which is 100 by 3. So, A is 100 by 3 where hundred is a number of measurements and three is the number of unknowns, all right.

(Refer Slide Time: 49:45)



So, our aim is to minimize the function J, the performance index; where J is equal to, what is the noise? See, it is like this; you are trying to fit in a cluster of observations, a best possible straight line, okay. I may choose this as the probably this the best possible straight line, there are many more measurements here and there okay.

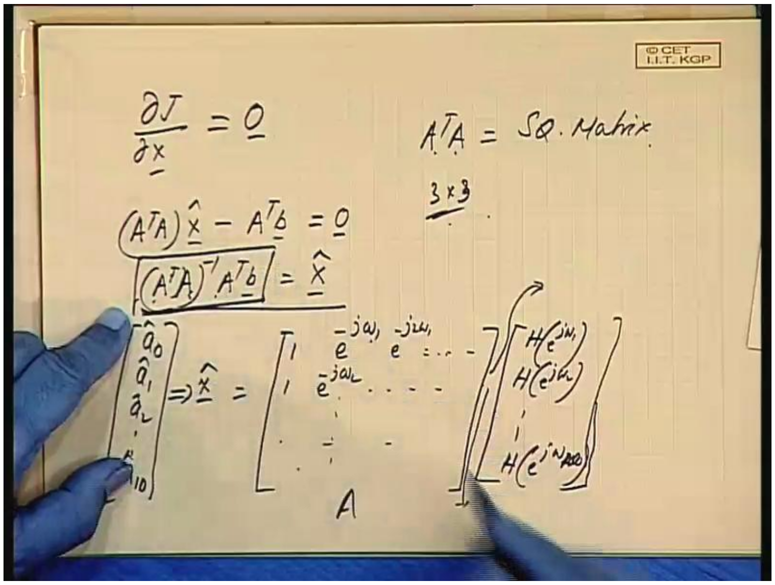
Suppose, this is the best possible straight line, how do you fit in that? I keep in changing the inclinations and then measure these errors, all right. So if this is x 1, this is the observed value; I assume some equation y is equal to m x plus c, I have to determine m and c, okay. So, y observed is this one okay, y at the i th point minus m x observed minus c minus c square will be this error. Summation of this, over I will be these errors squared okay.

Sorry and so on and this we try to minimize. So, similarly here it will be A x minus b, this error if this is an error vector; if error comes in the form of a vector all right, say e 1, e 2 e 3 then what is the sum of these squares? It will be e 1 square, e 2 square, e 3 square which will be E transpose E. So, if this is the error this transpose A x minus b will be the error, error square and this is to be minimized.

What I am try to minimize is e 1 square plus e 2 square plus e 3 square and so on. So, that gives X transpose, A transpose A X minus X transpose A transpose b minus b transpose A X plus b transpose b, okay. Now, this is to be minimized with respect to the parameters. Here basically parameters are the variable x, x 1, x 2, x 3. Here the parameters were m and c but they are the ones which are variable and these are the information, which are known y i and x i. So, parameters take the role of a variable, okay.

Now, here this is a quadratic, this and this they are same. It is like alpha is a vector; beta is a vector all right. So, alpha transpose beta is alpha 1, beta 1 plus alpha 2, beta 2 plus alpha 3, beta 3 is same as beta transpose alpha, all right. So, X transpose A transpose, basically this is A X transpose into b is same as b transpose A X. So, I can write this as two times any of these. And this is the constant, so derivative of this with respect to X will be 0.

(Refer Slide Time: 53:34)



So, I equate this to null vector which means; this will give me A transpose A is a square positive definite matrix, it is a square matrix. What will be the order of this A transpose A? See, A was hundred by three, so this will be three by hundred into hundred by three, so it will be 3 by 3 number of unknowns. If it is N, then it will be N by N matrix. So, it is a very compact matrix.

So, that will give me A transpose A, X minus A transpose b is equal to 0. Or A transpose A whole inverse A transpose b will be the solution; that the best possible solution, that we are searching for, we will call it optimum solution all right, best estimate. Now, what is our question in the filter design? Okay. We have got, now let us look back. We have got hundreds of such points all right. The information are given; the measurements have been made like b 1, b 2, b 3, b 4. So, these are the b's all right complex, the measurement vector is complex.

And we have to, we have to find out instead of three parameters say; a three variables x 1, x 2, x 3 now you have got up to twenty, that is twenty-one variables to be calculated. So, you can get the best possible estimates actually eleven, actually eleven. So, from here you will be getting the set of a 0, a 1, a 2 up to a 11. This will be taking the role of X vector X.

So, estimate of this we are making will be equal to capital A transpose A, capital A was; yes good you suggest, what would be capital A, what would be capital A? No, a 1, a 0 these are, what is capital A? 1 then e to the power minus j omega 1, all right, omega 1 may be 0; it may be or minus p i, whatever it is, I may start from this end then e to the power minus j 2 omega 1 and so on. Then 1 e to the power minus j omega 2 and so on, all right, basically it is very similar to Fourier matrix, all right.

Unlike, the Fourier matrix here you are having randomly selected, omega 1, omega 2 and omega 3 all right, they are not equally space. In the normal DFT, we are taking 2 p i by N as regular intervals; here I need not stick to that. This is also Fourier matrix of a different kind; it gives me the Fourier transform but not the discrete Fourier transform with regular intervals.

So this multiplied by, what will be this? H observations H at omega 1, H at omega 2 and so on, H at j omega 21 or 20 whatever you have taken. Not twenty-one, sorry 100; if you have hundred observations, hundred, is that all right? X will be now, that should be this one, this will be a 10 okay. And this one, X hat is equal to no actually; it is not complete sorry, it is not complete.

This is A matrix then you have to take transpose of that; then take the inverse of that, then again multiply by A transpose then multiply by this. There are some things in-between. I have written only the A matrix, basically you have to write this. It is the B column is this in-between the other two A’s are coming, okay.

So, we will stop here for today, in the next class we will be discussing in a little detail about the FFT algorithm. Thank you very much.