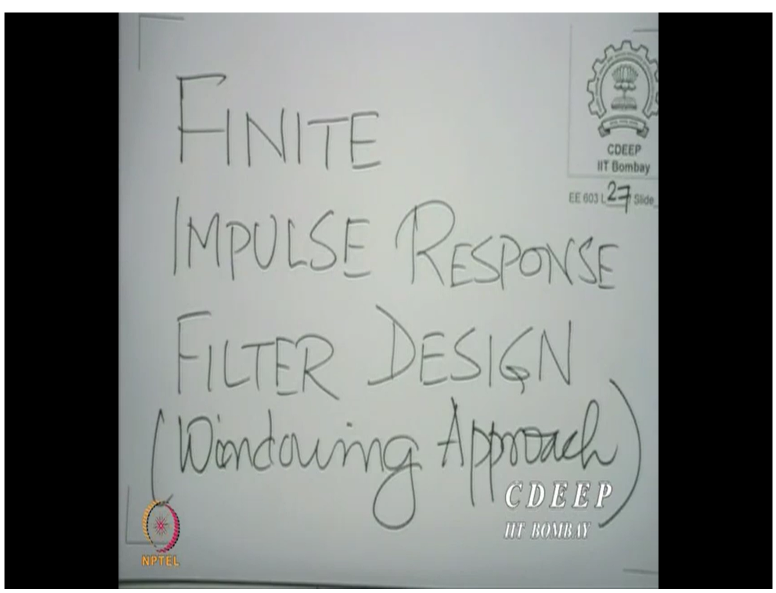
Digital Signal Processing and its Applications
Professor Vikram M. Gadre
Department of Electrical Engineering
Indian Institute of Technology, Bombay
Lecture 27 A
Finite Impulse Response Filter Design

A warm welcome to the 27^{th}lecture on the subject of Digital Signal Processing and its Applications. We have, in the previous lecture, completed the design of infinite impulse response filters using the bilinear transformation approach and frequency transformations. There are of course other ways to design infinite impulse response filters, that is not the only way.

Just to mention a couple, there is what is called the impulse invariant method. In the impulse invariant method, what one does is to keep the impulse response the same at chosen points of sampling, that is why it is called impulse invariant. And the impulse response of a discrete time filter around that principle.

In another approach, one can use optimization methods. So, one can use methods to optimize the coefficient with respect to a desired frequency response. We are not going to discuss those approaches in this course for want of time. Instead, we would now proceed to the design of finite impulse response filters and that shall be the theme of the lecture today.

(Refer Slide Time: 01:38)



We are going to talk about finite impulse response filter design. And specifically, we are going to talk about designing finite impulse response filters with what is called a Windowing Approach. There are various approaches, we shall look at a few of them based on the time that we have. But this is one of the most commonly used approached and therefore, we shall look at this approach in some depth.

Now, as the name suggests, a finite impulse response filter is one whose impulse response is finite in length. And obviously, the whole aim of that impulse response is again, to approximate the desired frequency response with as close an approximation to the ideal as one can get. Now, the design of finite impulse response filters can be likened in many ways to approximating numbers, except that this is one level higher. This is an approximation of a function and there we are trying to approximate just one number.

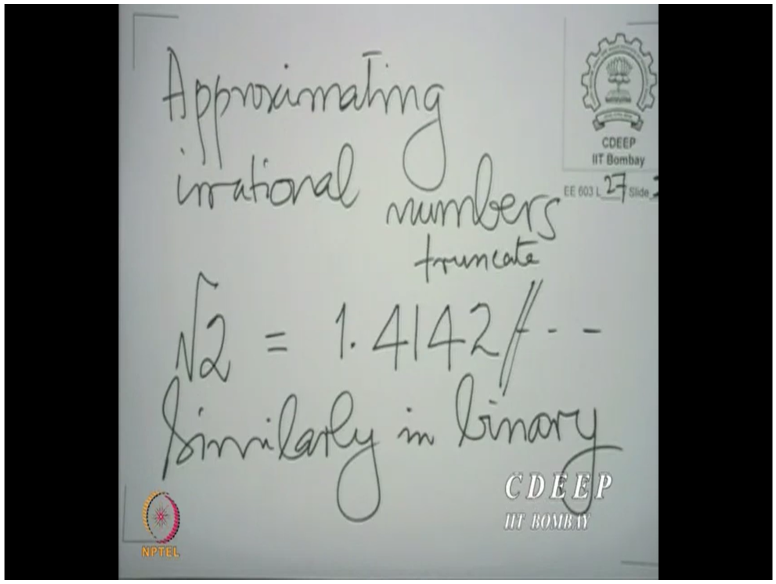
For example, suppose you wish to approximate the number $\sqrt{2}$. Now you know, the approximation of numbers and the approximation of functions, in many ways have parallels. In fact, one can take inspiration from the way one approximates irrational numbers. Now then, it is irrational numbers that really require approximation.

What I mean by approximation is approximation with a rational representative. When one carries out calculations in a calculator, which involve irrational numbers, one does not really use the irrational number at all in the calculation. One uses a representative which stands in for the irrational number and the representative has to be rational, it has no choice.

That is because you have only a finite number of bits in which you can represent any number, forget about irrational, any number in the computer and if your representation in the binary form exceeds that link, you are of course going to incur some error. But, what you are looking for is the best approximant, the best representative.

It is like saying that I have a function to perform, I have a task to do. And since, I cannot myself go and perform the task, I send somebody else who can do the task almost as well as I can, if not as well. That is what I mean by a representative. And some things to happen in the context of a finite impulse response filter. Anyway, let us look at how we would approximate numbers.

(Refer Slide Time: 04:51)



Now, I say irrational numbers because we have already seen that the ideal impulse response that we are going to deal with is irrational. By rational I mean, you can never realize it using a rational system function and we have shown why. So, what we are trying to realize is irrational but what we are going to use is rational. So, we have this conflict right in the beginning.

Now, in the IIR case, we resolve the conflict by allowing both zeros and poles. Now, we will not allow a denominator at all. The denominator would be trivial. And it is only the numerator polynomial which will play a role. And therefore, in a way, the design of FIR filters or approximating and irrational response by using FIR filters, is a polynomial approximation problem. It is a problem in which you are trying to approximate an irrational response by a polynomial.

A polynomial in z or z^{-1}, as you like it to be thought. Anyway, coming back, how would you write the number $\sqrt{2}$, if you were asked to do it in decimal? Then of course, many of us are familiar with what we call the square rooting algorithm but, let us assume that the first few digits have been calculated, so maybe you would have something like *1.4142* and something beyond that, the approximate $\sqrt{2}$. And of course, you can convert this to binary.

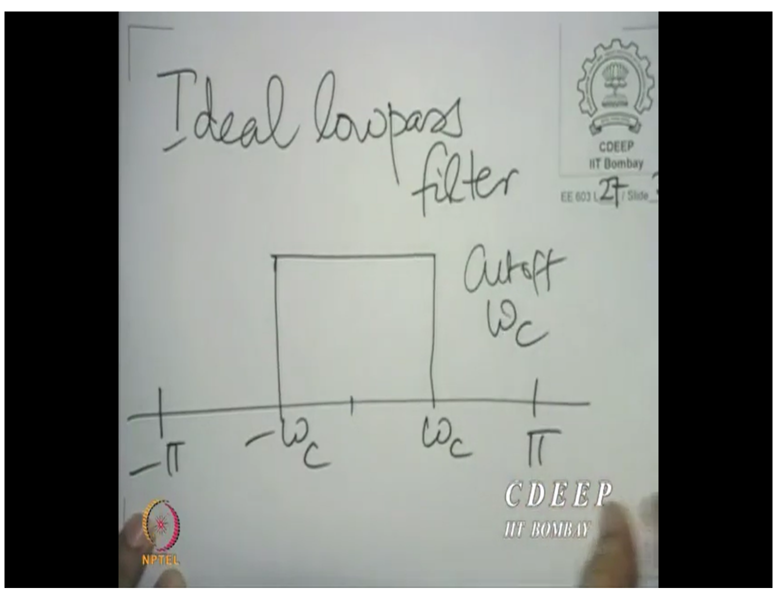
So, you see, what we do when we write this number on a computer is to truncate. That is one way to do it. You can truncate the representation up to a certain number of digits, not at all difficult to understand. Something better that we can do is to round it. By rounding we mean, we can look at the next digit and then we decide whether we should add *1* or not add *1* to the last digit that we have.

Rounding is, in some sense, better than truncation, because in rounding you are incurring only potentially half the error that you might in truncation. In rounding, you would not incur, you are at most *0.5* away, you cannot really do too much better than that. And the spirit behind rounding is that even after you have truncated, you are trying to see if you can, to some extent, put some band-aid on the wounds that you have created.

You see, by truncating you have actually cut off some vital part of that number. The number is not the truncated number, the number is much more than that. So, you have cut off something vital in that number. And that cut off or that process of severing those digits from the number is going to leave scars on the value of the number. What you are doing in rounding, in a certain sense, is to try and apply ointment on those scars in the best possible way and you cannot do too much better in the context of numbers and rounding.

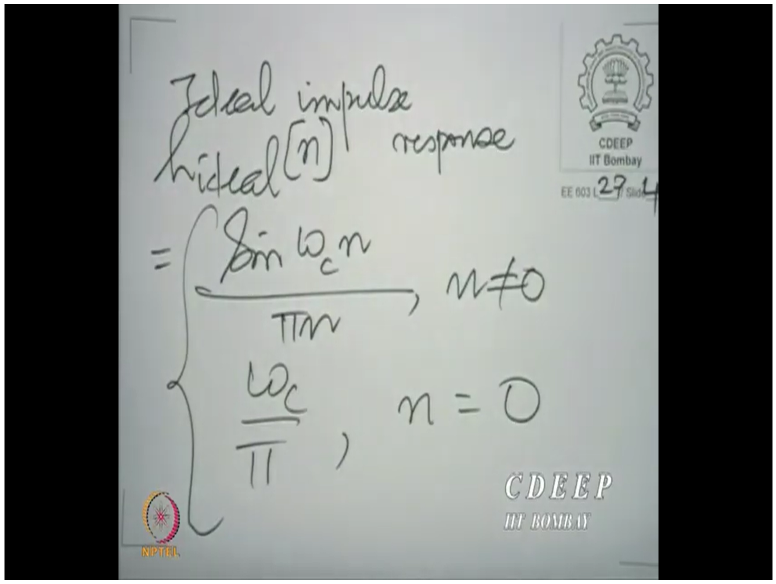
So, in the case FIR filters, one is doing something similar but at a slightly higher level. Let us take the example of a lowpass filter.

(Refer Slide Time: 08:46)



So, if you had an ideal lowpass filter with the following frequency response, we know what its ideal impulse response would be.

(Refer Slide Time: 09:09)

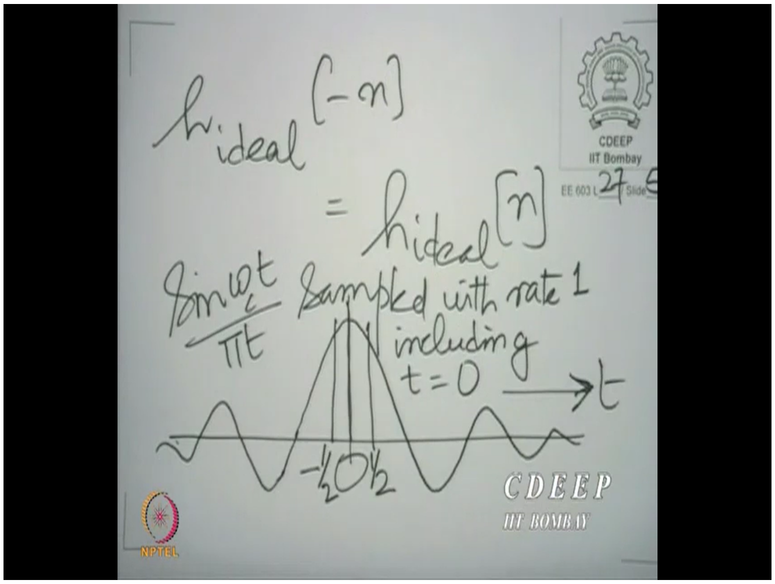


The ideal impulse response would be $\frac{sin\,\omega_c n}{\pi n}$, for $n \neq 0$ and $\frac{\omega_c}{\pi}$ for $n = 0$, we have seen this more than once before. Now, of course we are familiar with the fact that this is an irrational filter. So, it is also infinite in length in both directions. We have seen what precludes this from ever being realized. Let us just repeat. For one, it is infinitely non-causal. Secondly, it is unstable. Thirdly, it is irrational.

And if you want the finite length filter or finite link impulse response to go as close to this as possible, one obvious thing to do is to simply truncate this response. So, for example if you desire that you want an impulse response of not more than *11* samples in length, for the moment, I am taking an odd number of samples. And I will explain why.

You see, if you look at this impulse response, the ideal impulse response, let us call it $h_{ideal}[n]$.

(Refer Slide Time: 10:39)



So, if you look at $h_{ideal}[n]$, one very obvious thing about $h_{ideal}[n]$ is that $h_{ideal}[-n] = h_{ideal}[n]$. So, you see, if you look at the way you represent $\sqrt{2}$, so here, it is very clear that whatever you say on the positive side of *n* is mirrored on the negative side.

Now, if you look at the way you represent $\sqrt{2}$ on a computer, you retain the most significant places. So, suppose you were to calculate $\sqrt{2}$ up to *20* decimal places and you were asked, which of these decimal places do I retain. It is almost a no-brainer, you would obviously say that keep the most important ones. Keep the integer part and keep the next few places, as many as you can accommodate, does not require too much of thinking to come to that conclusion.

Now, in the case of the impulse response here, that is not so obvious. If I have the option of keeping *11* samples, which of them should I keep? It is not so obvious. Of course, intuition tells me that I should keep the most important of them. And in a certain sense, that indeed serves the same purpose as keeping the most important digits does. And what do you mean by the most important samples?

One way to identify the most important samples is to think of the samples with maximum magnitude. But then, you see, you cannot decide to keep some samples and throw away others in between. You are forced, if you do wish to truncate an impulse response, you cannot keep say, the first and third sample and throw away the second. That would lead to something very very peculiar.

So, you are restricted to of course, retaining or throwing away contiguous samples. And then, the most obvious thing to do is to look for that part of the impulse response which has the most important samples. And in this case, we can sketch how that impulse response looks. We know that this impulse response is a sampled version of the following envelope.

So, you can visualize this envelope, in fact, you also know what this envelop means $\frac{sin\, \omega_c t}{\pi t}$. And of course, at this point it is *0*. And so, what we have is $\frac{sin\, \omega_c t}{\pi t}$ sampled with rate *1* including $t = 0$. That is what you are doing. You have this ideal response, this ideal continuous function $\frac{sin\, \omega_c t}{\pi t}$, you are sampling this at every, at integer spacing and you are including $t = 0$.

Now, remember, you might also have decided not to include $t = 0$and we shall do that later. We are also going to give an interpretation to this continuous function that we have drawn. This continuous function is the band limited function that you would get if you restricted the discrete time fourier transform to the range $-\pi$ to π.

Now, remember, quite some time ago, we had talked about the underlying band limited function corresponding to a certain discrete time fourier transform. We said that the discrete time fourier transform is periodic with period 2π and if we choose, we can always restrict that discrete time fourier transform to $-\pi$ to π. What I mean by restricting is keep what is between $-\pi$ and π and throwout the rest.

If you did that, you automatically have a band limited function with a bandwidth of π and of course π on the normalized angular frequency axis. So, of course Nyquist Theorem tells us that we should sample at at least an angular frequency of 2π and that is all that is really about the discrete time fourier transform. So, when you sample that then, that means, you know, sampling in an angular frequency of 2π means a frequency of *1* and that means taking samples with an integer spacing and of course, then if you choose to sample it starting from $t = 0$, you get back the original sequence $h_{ideal}[n]$.

And of course, the fourier transform that sample sequence is essentially the discrete time fourier transform, maybe to within some constant. So, we do need this idea of the underlined band limited function here. In fact, the underlined band limited function as we see it, is symmetric in nature here, as you see, it is an even function of *t*.

And that is also reflected, now you know, it does depend on whether you have included $t = 0$ or not. If you have included $t = 0$ and then if you are sampling the integers, the evenness is preserved. But if you choose to sample say, starting from $t = \frac{1}{3}$ onwards and that evenness is not preserved, although the original underlying function is even, so one must be careful to see that evenness needs some effort to be preserved. There is another way to preserve evenness.

Instead of sampling at $t = 0$, you could have also sampled beginning at $t = -\frac{1}{2}$. So, you would get $-\frac{1}{2}, \frac{1}{2}, \frac{3}{2}, \frac{5}{2}$ and so on at the positive side and $-\frac{1}{2}, -\frac{3}{2}, -\frac{5}{2}$and so on so forth on the negative side. Even then you would have symmetry preserved, evenness preserved.

And you can see with some effort, that these are the only two ways in which you can preserve the evenness. Any other process of sampling of this with integer rate will not preserve evenness. Either take *0* and then take sample symmetry around and then if you do wish to retain that evenness after you have truncated. So, remember you have $h_{ideal}[n]$ which lasts forever on both sides.

But if you wish to keep the evenness of $h_{ideal}[n]$ after truncation, then you need to take an equal number of samples on the positive side of $t = 0$ as on the negative side. And therefore, we shall first begin with odd length finite impulse response design. So, for example, as I said, suppose you wish to preserve exactly at not more than, do not say exactly, not more than *11* samples in the impulse response then the most sensible thing to do, as you can see from here is to retain *5* samples on the positive side, *5* samples on the negative side and the sample in the middle at $t = 0$.

Now, on the other hand, if you are asked to retain only *10* samples, not more *10* samples and you still want evenness to be preserved, then you have to work a little harder. Clearly, you need *5* samples on the positive side and *5* samples on the negative side. So, then you have to resample this continuous function. You need to resample it starting from $t = \frac{1}{2}$.

So, let us look at this function. If you want an even number of samples, then you must sample starting from t equal to half and then you would of course include $\frac{1}{2}, \frac{3}{2}, \frac{5}{2}$ and so on. And $-\frac{1}{2}$, $-\frac{3}{2}, -\frac{5}{2}$ and so on. And then, you can take *5* samples on this side, *5* samples on this side and you would still have symmetry.

So, now it also tells us how to deal with a situation when you want an odd length FIR filter and when you want an even length FIR filter while preserving symmetry, while preserving evenness. For the odd length, there is no problem, just retain as many samples. For example, if you want an odd length filter of length not more than *11*, just retain *11* samples keeping the center intact, *5* on either side. If you want a length *21*, keep the middle, $t = 0$ and then take *10* on either side and so on.

But if you want an even length, then you have to work a little harder. You must shift the point of sampling by $\frac{1}{2}$ and then you take an equal number of samples on the positive and negative side. So, this should be observed right in the beginning, odd length or even length. So, now we will not keep on talking odd length even length every time, we will discuss the odd length case and for the even length case, you need to make that little change. And what is the effect of making that little change when you sample starting from $\frac{1}{2}$onwards?

All that you are doing is introducing a delay of half a sample, an effective delay of half a sample. It does not affect the magnitude of the frequency response, it only affects the phase response, that too it introduces a constant time delay at all frequencies, so it can be condoned. So, even length FIR filter incurs, I mean, even length FIR design by truncation incurs the additional penalty of introducing that delay which we anyway will see in a minute, that you cannot do without a delay. But you have to bring in that delay right at the beginning when you resample.

So, we will discuss odd length FIR filter design and for the even length, you just have to make this little change, you can keep that in mind right from the beginning. Anyway, now the point is that the simplest thing to do, as I said, if you wish to retain a certain number of odd samples, $2N + 1$, so let us assume that you wish to design...