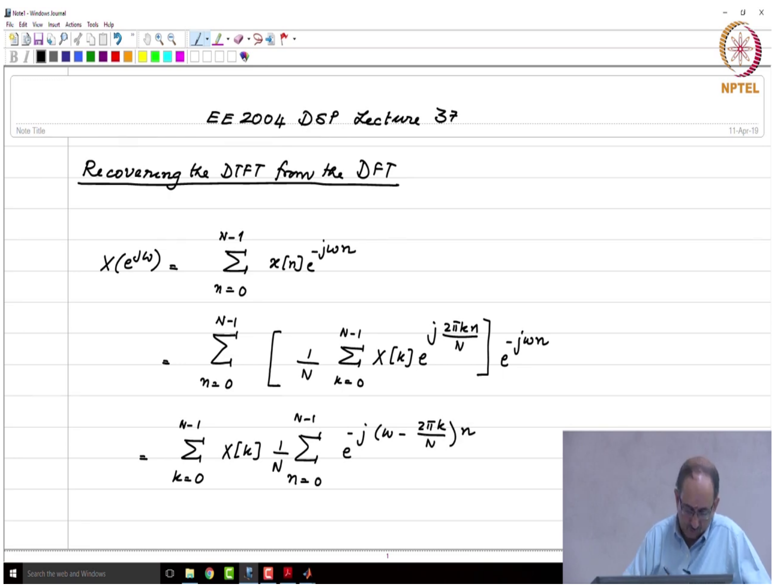
Digital Signal Processing
Prof. C.S. Ramalingam
Department Electrical Engineering
Indian Institute of Technology, Madras

Lecture 72:
The Discrete Fourier Transform (3)
- Recovering the DTFT from the DFT

So, let us continue with some of the other aspects of the DFT. Let us now talk about trying to recover the DTFT from the DFT samples.

(Refer Slide Time 00:32)



So, recovering the DTFT from the DFT. Again, this should make sense because we have an N point sequence and we have sampled it at N points in the frequency domain; that is, we have sampled the DTFT at N points. And if you sample the DTFT at N points, in the time domain it will repeat every N samples.

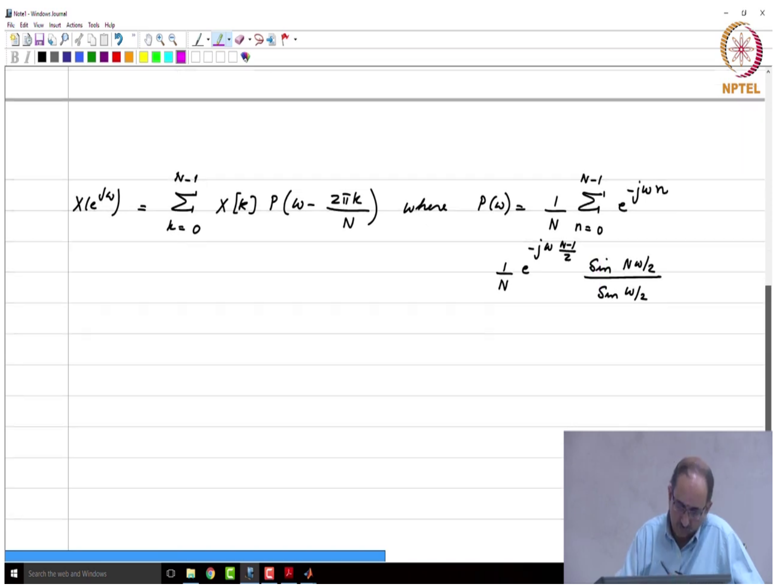
The signal is time limited, the repetitions do not overlap and hence there is no aliasing in the time domain. There is no aliasing in the time domain, from the frequency domain samples namely the DFT, you should get back the underlying DTFT; just like in sampling in the time domain. From the samples, you can get back to the underlying continuous time signal. And the way you did that was, you used an ideal low pass filter which corresponded to sinc interpolation.

You took the impulse train sampled signals and signal and passed it to an ideal low pass filter whose impulse response was $sinc(t)/T$ and that was sinc interpolation. Now, let us see what kind of interpolation comes about here and what are the similarities and differences. So, we have $X(e^{j\omega}) = \displaystyle\sum_{n=0}^{N-1} x[n]e^{-j\omega n}$.

Remember, our goal here is to relate the DFT samples and the underlying $X(e^{j\omega})$. Right now, you do not see the samples of the DTFT, namely you do not see cap $X[k]$ in this and to get $X[k]$ into the picture, you need to replace $x[n]$ by its inverse DFT. Therefore, this is nothing but $\frac{1}{N}\sum_{k=0}^{N-1} X[k]e^{j2\pi kn/N}$ and again there are no issues interchanging these two.

So, you have $\sum_{k=0}^{N-1} X[k](\ )$ and here you collect all the other terms, $\sum_{k=0}^{N-1} X[k]\ \frac{1}{N}\sum_{n=0}^{N-1} e^{-j(\omega-(2\pi k/N))n}$. So, this is all I have done is interchange these two summations.

(Refer Slide Time 03:52)



And, this can be written as $\sum_{k=0}^{N-1} X[k]P(\omega-(2\pi k/N))$, where $P(\omega) = \frac{1}{N}\sum_{n=0}^{N-1} e^{-j\omega n}$. And this of course is $\frac{1}{N}e^{-j\omega(N-1)/2}\frac{\sin(N\omega/2)}{\sin(\omega/2)}$, which we have seen any number of times before.

So, what is happening here is remember, on the left hand side, we have $X(e^{j\omega})$. So, what we are doing is, we are taking the samples $X[k]$ which are known only at spacings of $2\pi/N$ apart. So, these are the samples of the DTFT. From these samples, you are reconstructing the DTFT for all of ω, from for a continuum of values. And the way you are doing this is, you are interpolating them similar to what you did in the impulse train sampling case. There, the interpolating function was $sinc(t)/T$ as I just mentioned earlier, whereas now the interpolating function is this the Dirichlet Kernel.

And you will be able to easily see that 0, then $\omega = 2\pi k/N$, this will be, we will have a nonzero value. For every other value of k, this will go to 0. So, this is no different from what was happening in the sinc interpolation case. So, whenever you hit the sample value, contribution from other samples will be 0; only that sample will survey which is what should be because that sample value equals the value of the DTFT at that point.

So, you do not need any other contribution. It is only in between samples, you need contributions from every other sample. So, two things are different here; one obvious difference is the interpreting function now is this rather than your sinc t by cap T which is the analog sinc, the other important difference is that you can observe from the formula is?

Student: (Refer Time: 07:17).

Very good now can you.

Student: (Refer Time: 07:25).

Very good. So, this does not involve an infinite number of terms, it only involves finite number of terms. So, this is recovering the DTFT from the DFT. So, the interpolation function is the Dirichlet Kernel and only finite number of terms are used.