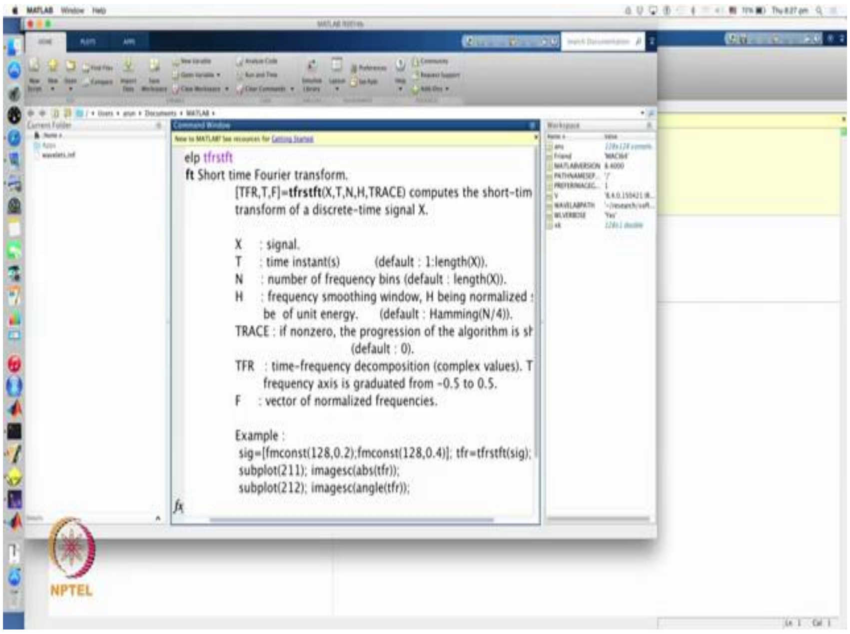
**Introduction to Time-Frequency Analysis and Wavelet Transforms**
**Prof. Arun K. Tangirala**
**Department of Chemical Engineering**
**Indian Institute of Technology, Madras**

**Lecture - 5.1**
**Auxillary**

(Refer Slide Time: 00:16)



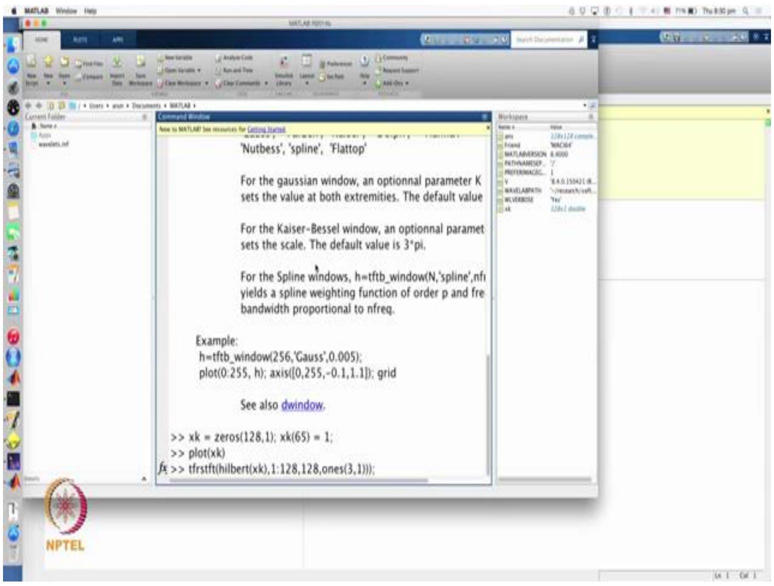

Let me show you how to perform the short time Fourier transform in mat lab especially using the time frequency toolbox, that is freely available on web. So, just search for time frequency toolbox and I have given you the reference you should able to download. Assuming that you have downloaded and installed, the installation is fairly straight forward. Now, and also you that you have put the time frequency toolbox on your mat lab path. All the routines that are available in a time frequency toolbox or now at your disposal. This is also a very nice reference guide and tutorial associated with this time frequency toolbox that you should read up.

So, let me get now to the point here, the routine that we are going to use is the t f r s t f t . t f r stands for short time, time frequency representation, and s t f t obviously stands for short time Fourier transform. It’s good to look up the help on this routine. So, that you know the syntax and the arguments that you can supply. The mandatory argument is a signal itself. Typically, it expects you to supply all the routines in time frequency toolbox, expects you to supply an analytic version of the signal that you working with. Whereas if you are working with wave lab then the there is no such requirement.

So, here the x is a signal that you are suppose to supply that is mandatory, the rest all are optional, for example time instants is the second argument, and the default values are given here, the number of frequency bins a quick word on the number of the frequency bins… Remember what you are doing in short time Fourier transform, you are slicing the signal. So, let say I am choosing 16 samples of a long signal, then what I am doing is, I am performing d f t of this 16 samples, and the d f t theory recall tells me that I need to at least compute the d f t at 16 frequencies. But I can also compute at much higher number of frequencies, and this is exactly the number of frequency bins that it is referring to at how many frequency bins you want to compute. If you compute at more than frequent number of frequencies more than the number of observations in the segment, you are just doing an interpolation, you are not actually creating in a new information.

So, that minimum value that you should supply here is the length of the window itself, but we will not touch it or even if you touch it will set it to the defaults, and the fourth argument is a smoothing window itself. So, the default is that it is uses a hamming window. We have not discuss the different windows, but hamming window has a bell like shape like the Gaussian one with a slight difference.

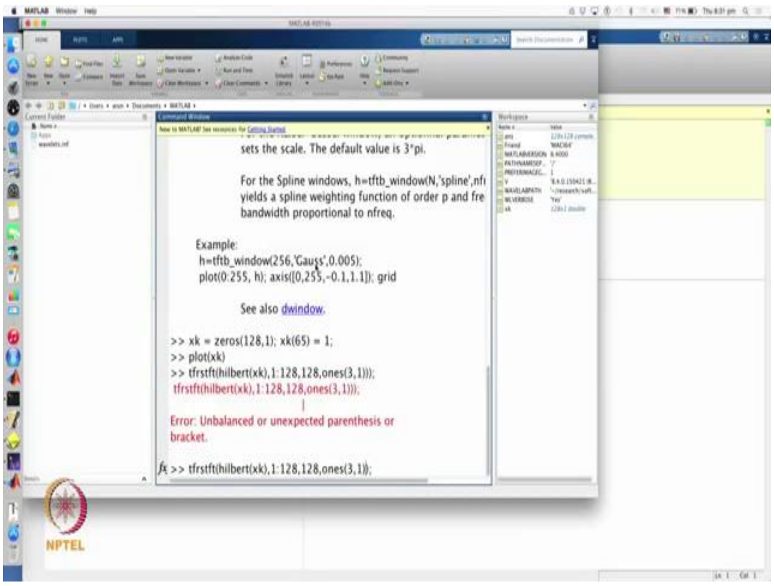
(Refer Slide Time: 03:08)



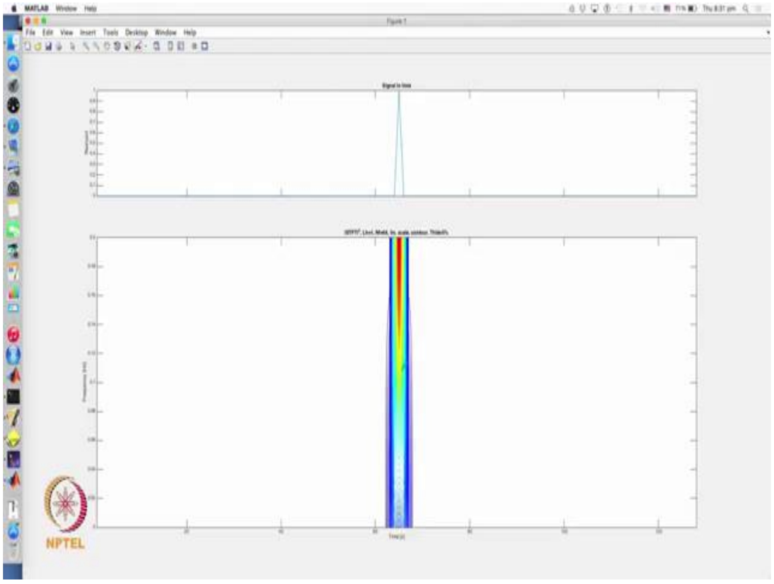
And you can generate the values of this window function using this routine call t f d b window, t f d b underscore window generates is window, there is also window function in mat lab, which we can generate the values of the window function. The requirement is that you supply a window function of odd length. So, with this background, let us run the t f r s t f t on two test signals; one is the impulse, and other is a sine wave. What I am going to do is generate the impulse here. So, I have generated zeros and I just had added a unity or in fact inserted a non zero value at the 65 th position, just to make sure that I have generate the right signal plotted, so it looks like an impulse perfect.

Now what I do is I supply the analytical version, analytic version - analytical version; analytic version of the signal using the hilbert routine in matlab, and I supply the vector of time instance, and I also say that I want to evaluated; these are all the default values and best explicitly specifying them. And I will leave the window function to the hamming or, because it is an impulse I can say I am going to use an rectangular window for example, of very narrow width. So, for this I supply a window function, ideally I can do one sample window, but let us not do that. I am giving a rectangular and slicing the signal with rectangular window of width three samples.

(Refer Slide Time: 04:49)



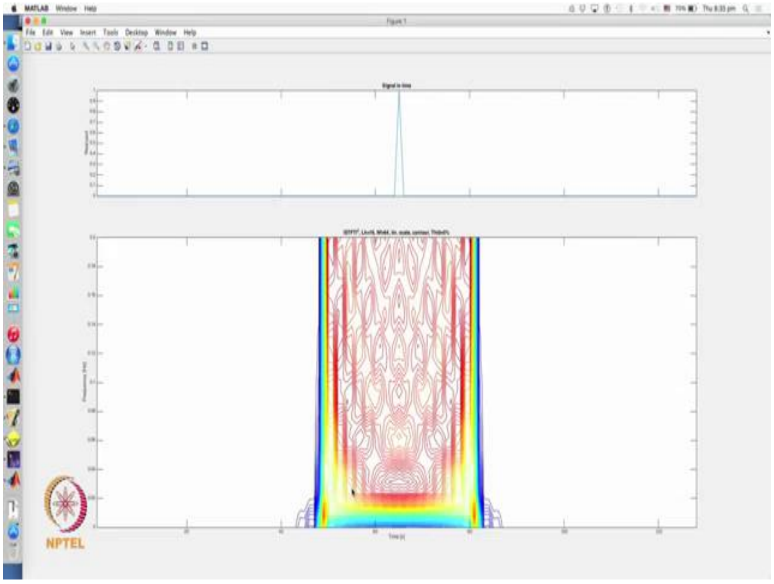
(Refer Slide Time: 05:02)



So, let me see what it actually gives me, fine. So, let me maximize the plot for you, here is the impulse ideally I should be plotting a stump plot, and here I have the spectrogram. In fact, we can improve this plot, the layout of this plot by choosing certain options here. I can change the layout, I can actually ask for the spectrum, and I can ask for the linear scale. So, I have done that. Now let us look at the plot. So, I have the signal on the top, the spectrum on the left, the spectrum make sense, because an impulse signal has all frequencies in it according to Fourier. And the spectrogram resolves this energy in the time frequency plane. It is says something is happening in this period. Of course, now look at the time spread of this energy or the spectrogram it is wider then actual time spread of the signal itself, that is again because of the window function that I have chosen. And of course, it is showing all frequencies you can have a color bar coming up showing you the values corresponding to these colors. Red - actually corresponds to the maximum intensity, and the blue corresponds to the weakest intensity. So, the color here essentially represents a magnitude of the short time Fourier transform.

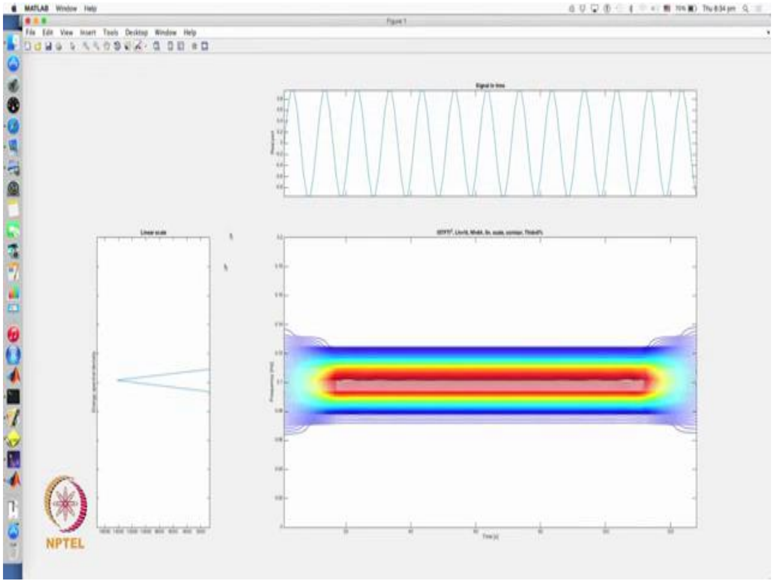
Now, what we can do is, just for the sake of it, we can choose a wider window and ask what happens? I am going to use a rectangular window, I can use a hamming window, but the question is a width of the window. Suppose, I choose a window of width 33 instead of 3, I have 33, let us ask what I see?

(Refer Slide Time: 06:48)



So, now you see what is happening? The spectrogram has lost, the time feature of the signal miserably. It's unable to localize in time, because I have chosen a wide window. So, now slowly you are actually going to get the Fourier transform, if you choose a much wider window, the entire spectrogram is going to be filled with colors, telling you that it each time there are all frequencies present in the signal, because that is of Fourier transform is viewing it. Notice that the spectrum is not shown here by default, but as I said you could again bring it up. In fact one of the things that you could do is you can say save options, and it says options saved. So, that in future plots of this t f r s t f t this spectrum is going to come out. And you can do many things you can set the lower and upper frequency bounds and so on. So, there is some beautiful options here that we should explore. Let's quickly ask how things look like, if I have a sine wave. I am going to generate a sine wave of discrete time frequency 0.1, and I am going to generate 128 observations of that, once again I invoke this. This time I will use this same window that I have used for the previous signal impulse and ask how things turn out to be.

(Refer Slide Time: 08:10)



So, as you notice the spectrum now it’s shown by default, because of save the options. This is exactly what is a illustration of what we did earlier on the board with respect the analysis of sine wave. The Fourier spectrum although it shows a kind of wide triangle, ideally it is a stem plot which means a Fourier analysis correctly picks a sine wave. The short time Fourier transform yes, thus say that the sine wave exist throughout the analysis the period or observation period sorry, but its unable to tell you what is exactly the frequency present in the signal, and this is what we say is a smearing of the energy of the signal. The true energy of the signal is actually truly localize at a single line at is at each time it is the single frequency, but if I peak any time interval, and if were go by the spectrogram, I am let to believe that there are all these band of frequencies present in the signal which is not the case.

So, you have some kind of misleading information here, and that is because the energy of the signal and the window of scramble, they are actually mixed up jumbled up, and that is why you have the situation. Of course, the maximum intensities present here which is over a much shorter bandwidth, band of frequencies then what you see here. So, you can say that this energy of the signal is encased in the energy of the window itself. I can improve this by choosing a wider window, and I leave that exercise to you, but the question is what happens? If I have both sine and impulse, and we will evaluate that those kind of signals later on particularly when we study the practical aspects on window length, and window tight, and so on. This is just a preview and also to get you started on out on the practical aspects of the short time Fourier transform, particularly the computational aspects. So, will meet again in the next lecture where will talk about the theoretical properties of the short time Fourier transform.

Thank you.