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

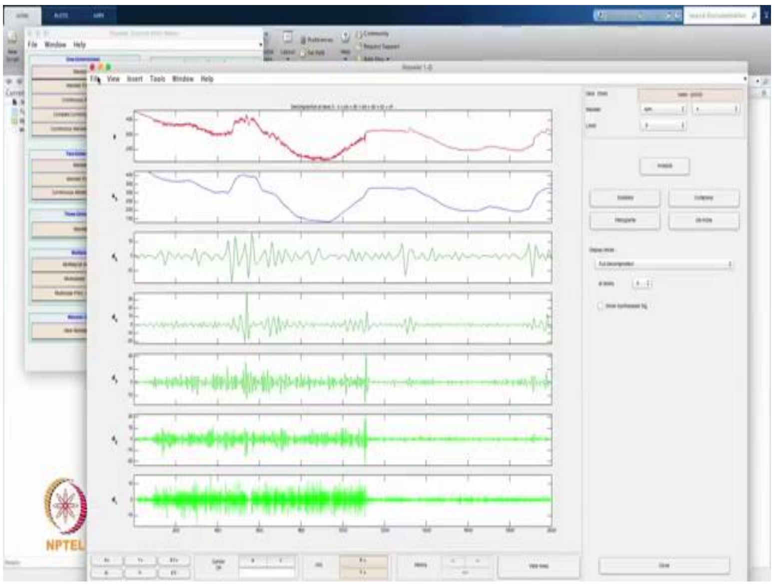
Lecture – 08.5C

Let us look at an example here, where we are looking at the possibility of applying different thresholds to different intervals of coefficients. In the previous example, we looked at the case of applying a single threshold, the universal threshold. And as I said is, the universal threshold only requires the estimate of sigma. And you can use any estimator of sigma. The standard estimate as I said is the square 2 norm of the detail coefficients.

There exist many other estimators of sigma all it all depends on, how you look at it. For example, I can look at the median of the detail coefficients and use the correction factor to that median. That correction factor is usually point 6745. So, I can say sigma had it is median of the detail coefficients divided by point 6745. I have that in lecture notes for you, you can look up that.

Then, there exist many other ways, call mini max way of estimating the threshold, which gives you tables, for values of sigma depending on the signal, the characteristics of the measurement and so on. Now, we look at this example, where I may have to apply two different thresholds to two different intervals of the coefficients.

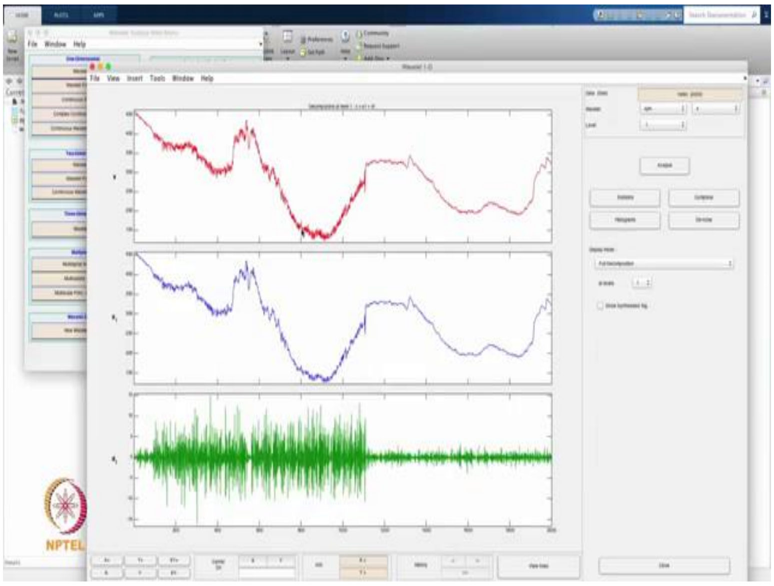
(Refer Slide Time: 01:44)



And one such example here is given, it actually comes loaded with mat labs wavelets tool box. So, go to file, example analysis and choose here, noisy signal. And you can see interval de-noising. And choose the last one here, electrical consumption. Ignore all the plots, except the top one here. The top one contains a signal itself. And you can see clearly that, the noise levels are different as you move from this first half to the second half.

In fact the second half has much lesser noise in it. So, to be able to see this, let us choose a one level decomposition. So, that I have more of the signal shown.

(Refer Slide Time: 02:33)

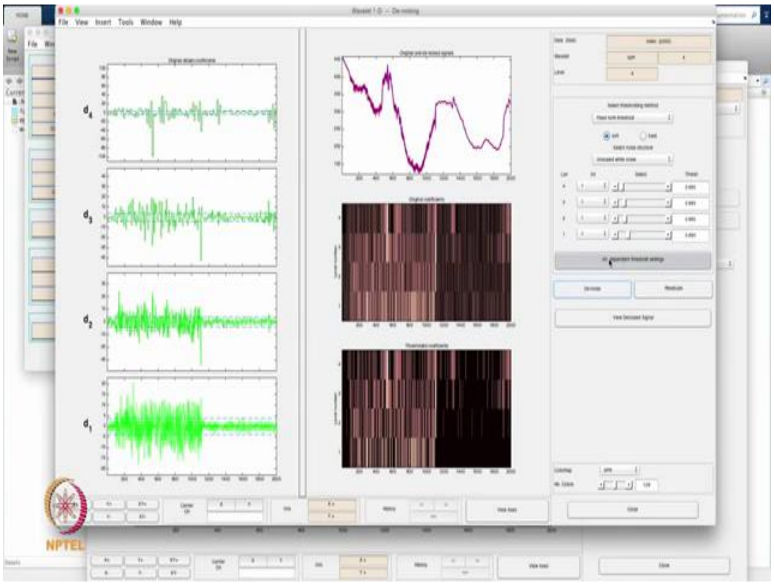


Look at the signal here. Now, you can see clearly. This is the one level decomposition, due to hour. It is always good to begin with an hour wavelet decomposition. Because, it gives you good time resolution characteristics one. And secondly, it does a good. It is a default difference filtering kind of option. So, it gives you decent estimate of the noise levels and so on. So, if you look at this decomposition, the top is a signal.

The middle plot is the approximation, reconstructed approximation and the bottom part is a reconstructed, details that are left out. So, if you sum up the blue and green, you get the red. You can clearly see that, even if you simply assume that the green one is the noisy component of the measurement. The noise levels are totally different, beyond the certain point. In this case, applying a single threshold to the entire set of detail, coefficient at any scale is not the vise think to do.

So, we can choose a different wavelet also. Here, I think the stimulate wavelet is being used. So, we will leave the wavelet as is. But, we will change the level of decomposition, may be to a four level decomposition and then ask for de-noising.

(Refer Slide Time: 04:01)



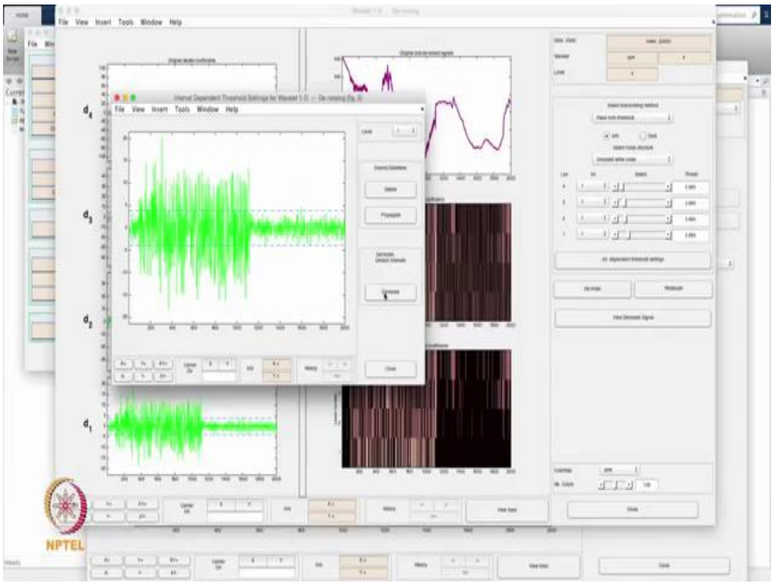
And here, we shall of course to just to see the difference between the interval level thresholding, interval based thresholding and a single thresholding. First, you should apply the single thresholding. And this thresholding again is indicated on the left with the blue dash lines. And we can ask for de-noising using the soft thresholding approach. Here, I do not have any the short features. So, I can still use a soft thresholding.

It will produce a soft estimate. What you see as a color map plot here? It is the color map of the coefficients plot, which is… I am in it, basically again telling you. In fact these coefficients are only detail coefficients. It is once again telling you that, there is more energy contained in the first half of the time period versus the second half of the time period, which is the darker. The darker color here corresponds to lower energy.

So, now if I simply use the standard same fixed threshold over the entire scale and de-noise, then I get such a signal here. So, this is the approximation. But, if I use an interval based thresholding, then what do I do? So, what I do here is, I say generate the intervals automatically. So, I can specify visually by looking at the coefficients visually. I can specify, how many intervals of noise variations are present in the data.

Here, I have two intervals. In the first half, the noise level is much higher and the second half. But, I can ask it to generate automatically.

(Refer Slide Time: 05:55)



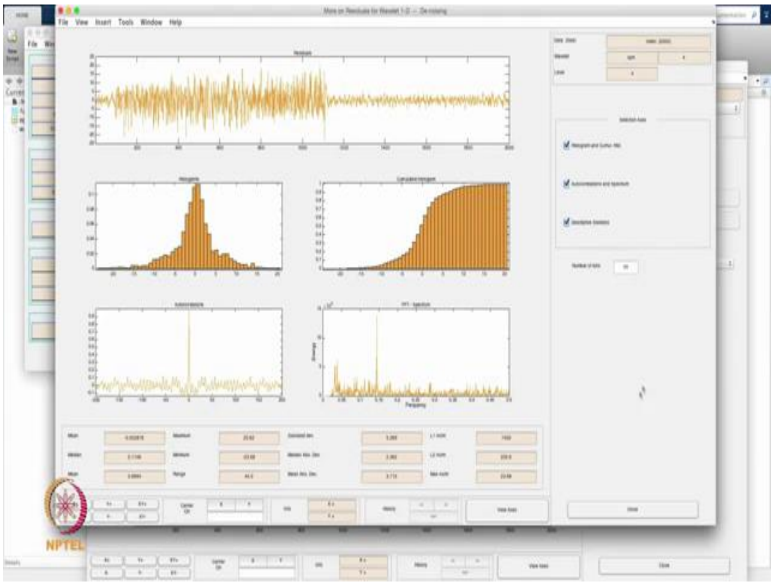
So, when I say generate, it automatically figures out that, there are two intervals in the data, where the noise levels are differing. And that is indicated by the red dash line here. And now, I can say use this. So, let us choose here two intervals. And then, say apply. So, update. Now, you update and what it has done, as you can see in the sliders here. In fact, you can manually adjust the thresholds. It has actually chosen different thresholds also and for different intervals.

And now, you can ask for de-noising. This is the de-noised version. So, you can look at the bottom plots here. The bottom plots here, is showing you how the coefficients look like after the thresholding has been applied. Here I have applied an interval threshold. On the top here, I can see the deconstructed signal in blue. It looks much better than the one that I had obtained, by applying a fixed threshold. So, let us actually go back.

And I do not know. So, just go back as an exercise in the interest of time and I am not going to do it. So, just go back to the case, where we are applied a single threshold and compare. In fact, all of these can be exported. So, if you go to file and here when you say save, you can save the de-noise signal. You can save the coefficients you can save the decomposition and so on. So, when you say, de-noise signal, it will ask you to save it in a file and so on.

So, you can of course do many other things. But, hopefully you have seen the difference between the signal that, I have obtained by applying interval thresholding versus the fixed thresholding.

(Refer Slide Time: 08:35)



It is also useful to look at the residuals from your de-noising. That is the final thing that, I want to show you. And that will give you an idea of, whether you have performed the good signal estimation or not. So, what you see on the top here, how did I come to this plot here? The way I have come to this plot is by hitting the residuals buttons here. And that bring up the residuals. It gives me a histogram. It also gives me, what is known as the auto correlation function.

If the assumption, normally what I assume is that the signal has a white noise in it, the measurement has a white noise in it. But, the measurements can also have correlated noise. But, suppose I have assume right noise and I perform estimation and I have a made a mistake, then the residual analysis will reveal that such discrepancies. So, here what does a residual analysis show me? It shows that, these are the residuals at the top, which is good.

Because, it is showing clearly that it has extracted the noise, qualitatively at least correctly. And at the bottom, I have the histogram. The probability distribution plot indicating that, it has more or less a Gaussian distribution. And what you see at the bottom here, below the histogram plot distribution plot is the auto correlation. If the signal is white, the auto correlation should have a single spike at lack zero and insignificant values at other lacks.

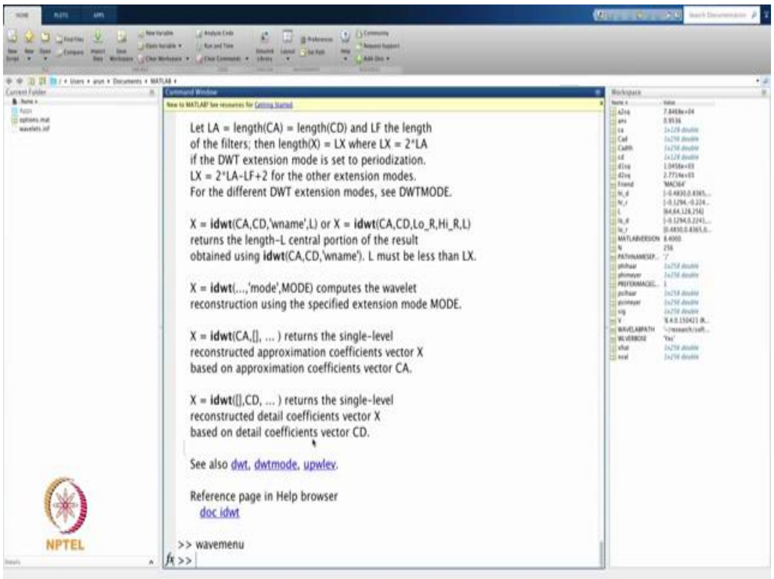
And that is, what it shows here. On the x axis, you have lacks. Hopefully, remember what is auto correlation? On the right hand side, I have power spectrum. And if the residual has white noise characteristics, then the power spectrum should be more or less plot. I should not see a significant trend. All of this is more or less satisfactorily met here. So, I would like you to take an example, where you have correlated noise and perform a signal estimation exercise and look at the residuals.

And it will show you clearly that, you have made a wrong assumption of right noise. Then, what happens if it is correlated, then you has to use different thresholds. The universal threshold that we applied by default. With all the default options, it is good for white noise. For correlated data, that is correlated noise you have to use different thresholding characteristics and so on. There is so much of literature on this.

That it is obviously impossible for me to go over in such a short span. But, hopefully the basic ideas have been conveyed. And what I had promised to you is that, I will give you references which discusses many of these different situations and does a very nice survey of the different methods. So, play around with this GUI. There is so much to learn from this GUI. And the plots are also quit useful and you should play around with the different plot and so on.

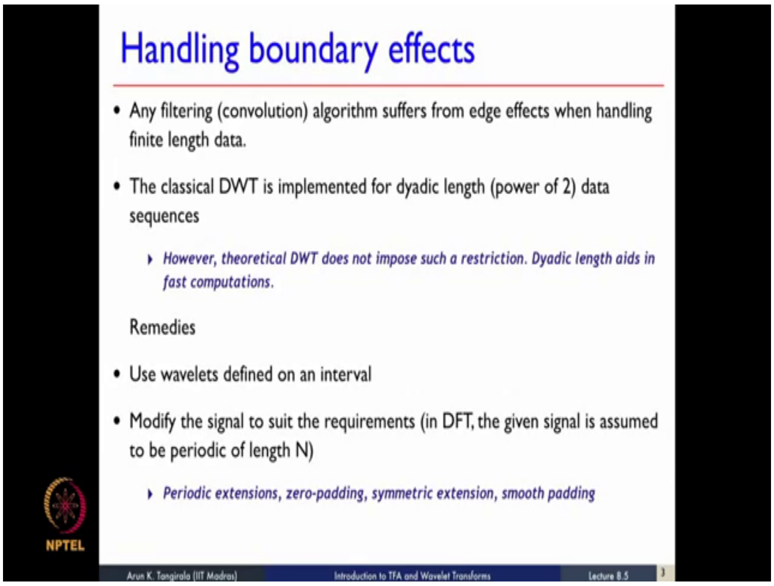
There are advanced options that you can exercise to generate the plots that are useful to you, in your application.

(Refer Slide Time: 11:54)



So, I going to close the GUI here and quickly take you to the final lecture on this topic, which is essentially summarizing, what we discussed. Except for one point, which we have not discussed, which is how the wavelet implementations in mat labs tool box. Or in general, handle the boundary effects. I am just going to talk to you briefly , the boundary effects for 5 to 7 minutes.

(Refer Slide Time: 12:17)



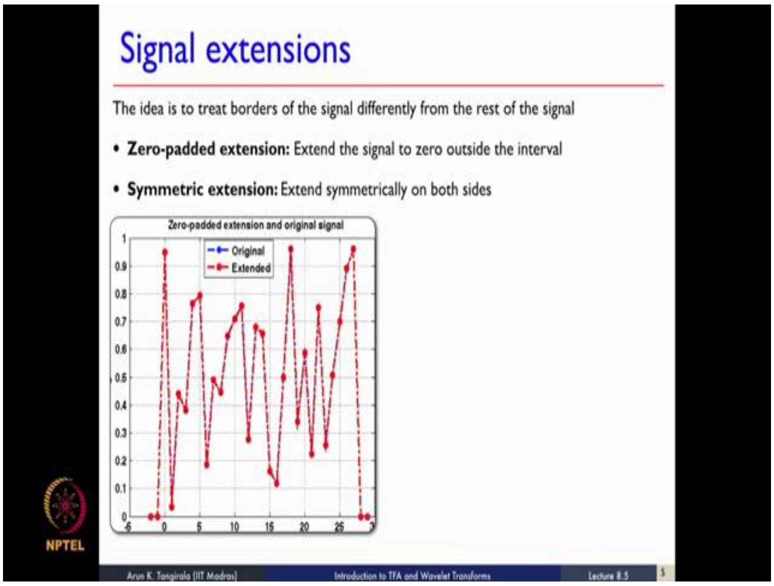
So, as I just said any wavelet analysis, the DWT or CWT involves convolutions. And convolutions always suffer from boundary effects. And how do you handle boundary effects? Well, the standard solutions that exist in a filtering literature are used. Either, you assume that, the signal is periodic outside the interval. Why do this boundary effects even arise in the first case?

Because, remember when you place the wavelet or the scaling function at the start of the signal or at the end of the signal, there is a portion of this scaling function or the wavelet, that lies outside the observation interval. And that is why, you end up with this boundary effects. And we are also studied, cone of influence for example, in CWT. That is also an effect of the boundary effect. So, coming back to the solutions, there are many solutions.

One is that, you could periodize. That is, you can have periodic extension of the signal, outside the interval or zero pad or do a symmetric extensions or smooth padding. These are the four different solutions. There is another solution which you can also pursue; you can read out the literatures, which are called wavelets on an interval. That is, you use wavelets exactly, that are defined only over the interval of the signals extension.

And that is finally, theoretical. But, they are used also in implementation. Largely, we will only discuss these four possibilities, very quickly. We know, what periodic extensions and zero padding’s to they introduce artificial discontinuities. And symmetric padding is much nicer or to the signal, at the boundaries. And then, there is a smooth padding which is essentially and an extra collation of the signal, at the boundaries.

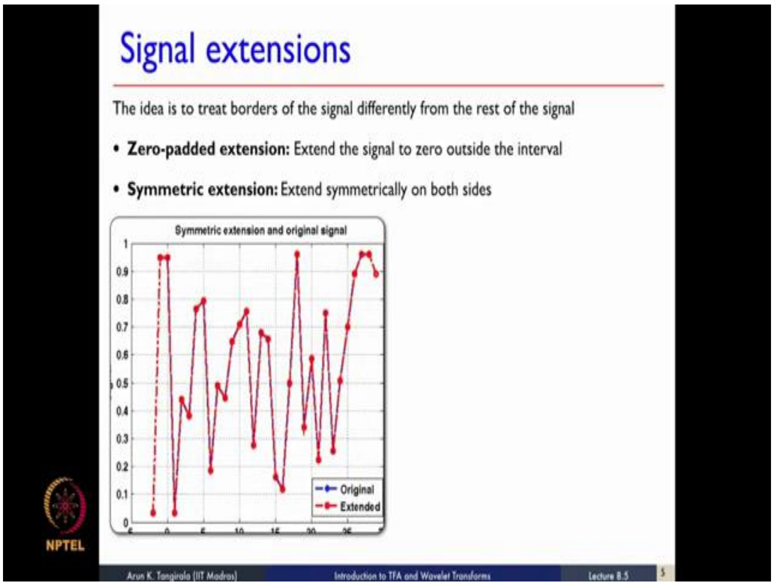
(Refer Slide Time: 14:12)



So, let us look at, what is the zero padded extension. How this signal is extended using the zero padding? So, the one that is shown in the red is an extended version. And that one that is shown in the blue is original. You do not see the blue signal, because over the interval of observation. That is, from 0 to 28 you have the signal 27. And then, what I am doing is, I am performing an extension of the signal, outside the interval because of the filter length.

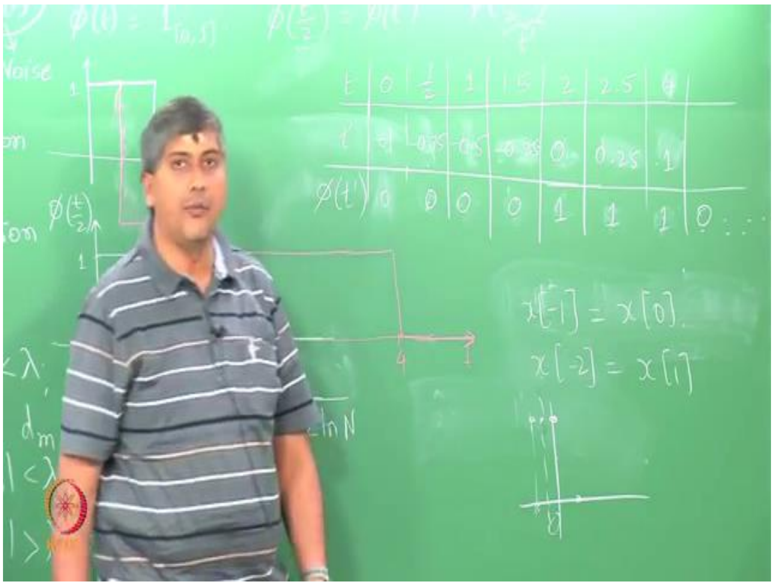
How much do you have to extend the signal outside the line? Outside the observation interval depends on the length of the filter. If I am using daubechies two filter, which has four coefficients, then I will need extensions to the left of how many I will need, two to the left and two to the right and so on. So, here I am assuming that I need only one extension to the left and right. This is only for an illustration purposes.

(Refer Slide Time: 15:23)



If you have to do zero padding, this is how it would look like. And if you where to do a symmetry extension, as you can see to left and right I have symmetric extensions, whereas left to the left of 0 and right is to the left of 27. So, you have… I have done here, two point symmetric extensions. And this is known as the half point extension. There is something called half point extension and a whole point extension. The difference is very clear.

(Refer Slide Time: 16:06)



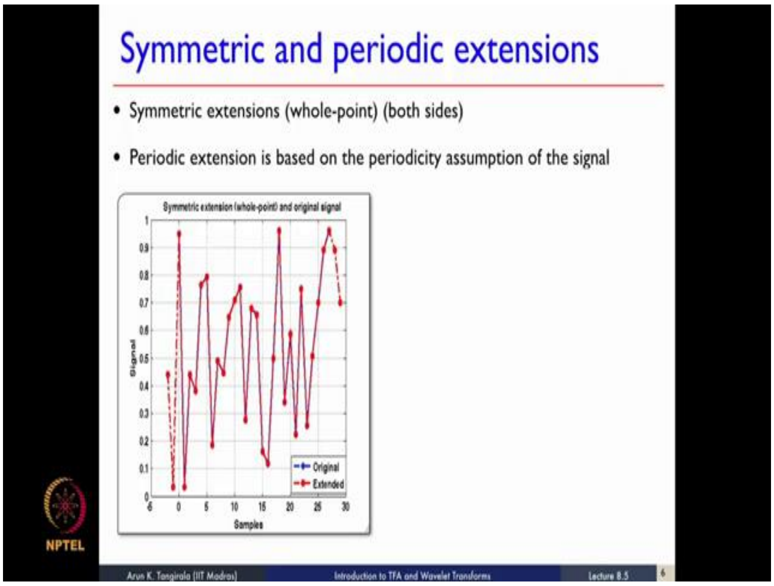
In the half point extension, let us say I need to generate x of minus 1. What I would do is, I would set that to x 0 and then x of minus 2 to x of 1 and so on. This is half point extension. Why is it called half point extension? Because, the point of symmetry is between minus 1 and 0. See, the point is where you think, the symmetric is. The signal begins here. Let us say, the signal begins here at 0. If you look at signal that, I have here, the next point I have here.

So, I need values at minus 1 and minus 2. Where is the point of symmetry, if I am doing a symmetric extension. If I am adopting this extension here, what I am going to do is at minus 1, I am going to set the values to x 0. By doing so, what I am doing is I am assuming that the point of symmetry is here. This is the symmetry line. That is why it is called as half point extension. The symmetry as I set, minus half.

In a whole point extension, as I show you what we do is, we set x minus 1 to be 1 and x minus 2 is to be 2. Whereas assume, the symmetry point is that 0. So, in a whole point in symmetric extension x minus 1 would be simply this value here. So, which extension should I use? If I am using even filters like daubechies 2 and so on, there are all even filters, sense even number of coefficients. Then, it is recommended that we use half point extension.

And for hard length filter coefficients, the whole point extension is recommended. This is nicely discussed in the book by Hilbert’s time. And again, it is known as wavelets and filters bang. It is a very celebrated book.

(Refer Slide Time: 18:04)

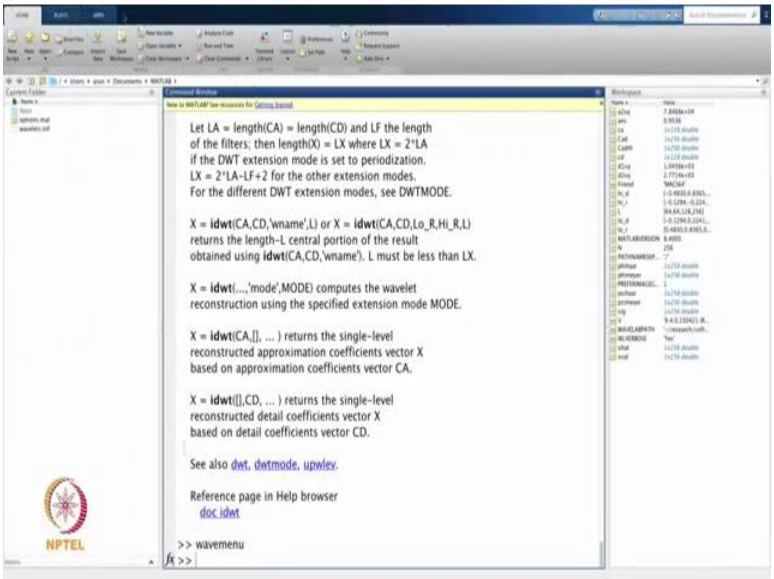


So, I just showed you in the previous light half point extension. Now, the whole point extension on both sides as I said, look at it. At the ends, the repeated values that is, if you look at the point 0, then you have minus 1. At minus 1, what is the value? It is the value at 1. Whereas, let me take you back to the half point extension. This is zero padding. Look at half point extension. At minus 1, the values are at the 0 itself. That is, what is the half point extension.

So, hopefully you understood. And there is no need that, you have to extend both sides also. Depends on what you are doing, you can extend on one side alone. Periodic extension, of course is based on the periodicity assumption. You simply assume, the signal to be periodic. Now, both periodic assumption and the zero padding extension introduce artificial discontinuities and they can bring about a lot of furious artifacts.

It does not mean, other extensions do not, but they are much nicer at the signals. So, symmetric extension is a very good option. And if you are using as I said, even filter coefficients, the half point extension is what is used. In fact, if you go to mat lab and you want to know, what extension mat lab uses, then simply go to mat lab and tight DWT mode.

(Refer Slide Time: 19:27)

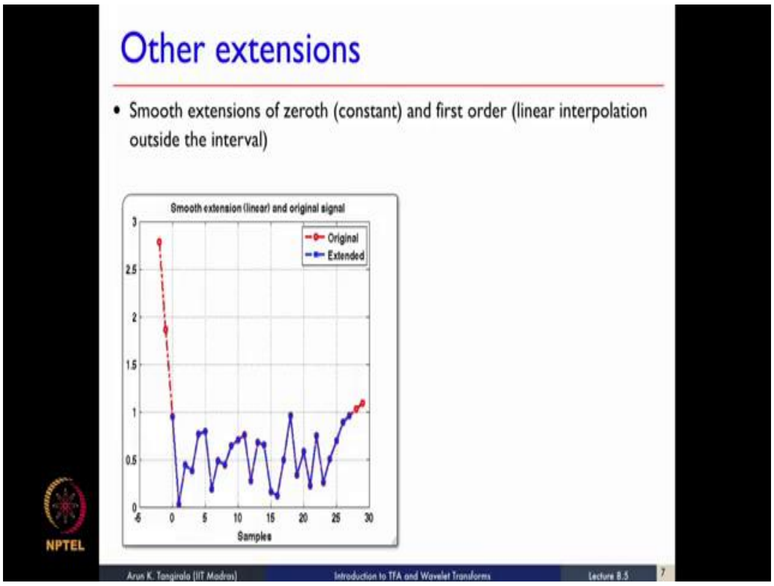


So, it uses a symmetry half point extension. Why does it uses, because most of the filters that we are using are even coefficient filter. And therefore, you can uses. But, you can choose for an example to… So, if you look at help DWT mode, it will tell you what are all the extensions that are possible. You say, it is the symmetry and then you can whole point symmetry or you can use an antisymmetric extension, half point and you can use an antisymmetric.

You understand, what is antisymmetric as suppose. It is actually x minus 1 would be not x 1, but an antisymmetric manner minus x. And then, you have zero padding option. And then, you have the periodic padding, smooth padding and so on. So, DWT mode pad sets the DWT mode in a periodic mode and so on. So, there are different options. And it is given here. The default mode is loaded from the file DWT mode.

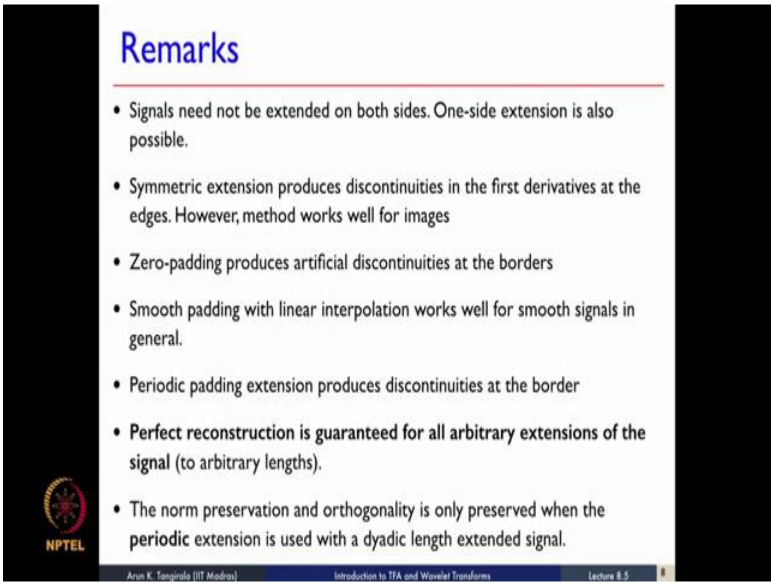
You can go and change, if you want. So, that your DWT mode is always, that is extension mode is always to the one, that you desire. But, it is better to retain the symmetric half point extension as long as you are using the even coefficient filters. So, let us quickly conclude this presentation. So, this is a symmetric and this is a periodic extension. And then, there are other extensions as I said. Finally, smooth extensions of zeroth, that is constant.

(Refer Slide Time: 21:11)



What you assume is, the signal is constant outside the interval. That is called as smooth extension of zeroth order. The zero signal was constant before and after, to the values at x 0 and at x n minus 1. First order would use interpolation. That is, it uses x 0 and x 1 to calculate the slop and extended outside. And at the other end, it would use x n minus 1 and x n minus 2 to get. So, now which is good, which is bad?

(Refer Slide Time: 21:47)



Let us talk about that, very quickly. The symmetric extension produces discontinuities in the first derivative. Therefore, if you dealing with signals itself and you not worried about the derivatives, symmetric extensions are very good. Despite, introducing discontinuities in the first derivative at the edges, it still works well for images. Because, in images mostly I am worried about edge detection or equivalent of discontinuity detection in signals.

Because, it introduces the discontinuities in first derivative, it should work well even for pure edge detection. Zero padding as we are mention, it produces artificial discontinuities. You should not do it as much as possible. You should not, do not resort to zero padding at all. Smooth padding works well for smooth signals. If the signal is smooth, definitely an interpolation would be really nice.

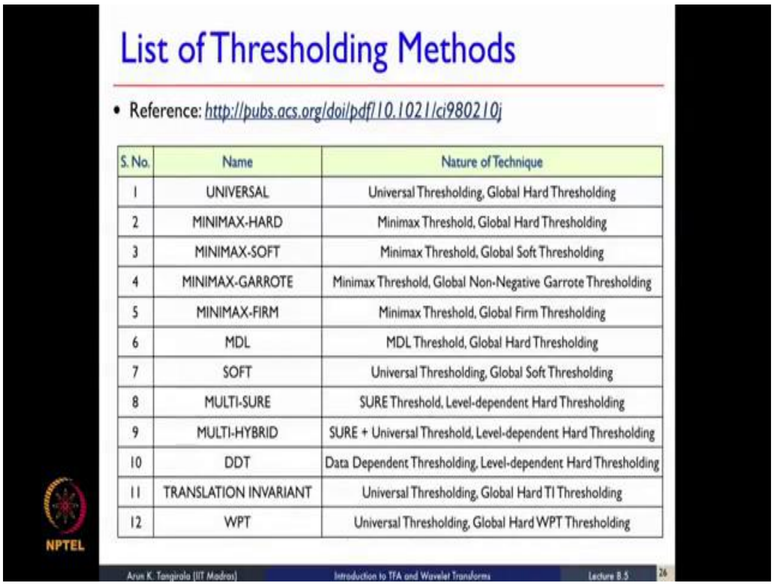
And the level of smoothing, the order of smoothing, whether you are going to use first order second order and so on, depends on the signal. If you know a priory, what this signal is before. Once again periodic padding introduces discontinuities. Now, regardless of the extension, the perfect reconstruction is always guaranty. In fact, what you should try is, in the example analysis that I have done. I would taken a signal of length, which is power of 2.

But, what it should do is, take a signal of arbitrary length and see, what extension the signal performs. I leave it as a simple exercise for you. And then, see if you are able to recover the signal, despite the extension that is being applied. And all, this is what is important. If you want the norm preservation and the orthogonality to be preserved, the periodic extension is the only one that guaranties the norm preservation.

What is this norm preservation? Earlier we said that, the sum square signal is the sum square of the approximation and the detail coefficients. That is theoretically guaranteed, only when periodic extensions are used. But, it did work for the signal that we used, although this is a default mode symmetric half point extension. It works, but that is probably a case specific thing and probably, because we are using a length which is power of 2.

But, strictly speaking the norms and orthogonality are preserved, only if you use periodic extensions. So, finally let me give you the reference for the signal estimation.

(Refer Slide Time: 24:36)



S. No.	Name	Nature of Technique
1	UNIVERSAL	Universal Thresholding, Global Hard Thresholding
2	MINIMAX-HARD	Minimax Threshold, Global Hard Thresholding
3	MINIMAX-SOFT	Minimax Threshold, Global Soft Thresholding
4	MINIMAX-GARROTE	Minimax Threshold, Global Non-Negative Garrote Thresholding
5	MINIMAX-FIRM	Minimax Threshold, Global Firm Thresholding
6	MDL	MDL Threshold, Global Hard Thresholding
7	SOFT	Universal Thresholding, Global Soft Thresholding
8	MULTI-SURE	SURE Threshold, Level-dependent Hard Thresholding
9	MULTI-HYBRID	SURE + Universal Threshold, Level-dependent Hard Thresholding
10	DDT	Data Dependent Thresholding, Level-dependent Hard Thresholding
11	TRANSLATION INVARIANT	Universal Thresholding, Global Hard TI Thresholding
12	WPT	Universal Thresholding, Global Hard WPT Thresholding

Here, this is the reference to the paper that discusses different thresholding methods. And you can see a big table here, which has 12 entries in it. The name derives from the kind of method, you are using for the thresholding. And how you are applying the thresholdin?. So, universal thresholding is a default one. And this third column tells you, how the thresholding is being applied, what universal means in their paper?

This is not necessarily, the Norman cleavage that is used in the wavelet literature. But, in that paper what universal means is, the universal method thresholding is used to calculate the threshold. And it is applied globally, which means to all scales. And how it is applied in a hard thresholding manner. So, if I pick for an example mini max soft, which is the third entry. Then, a mini max method, which has a table of thresholds for different situations that is how the threshold is computed?

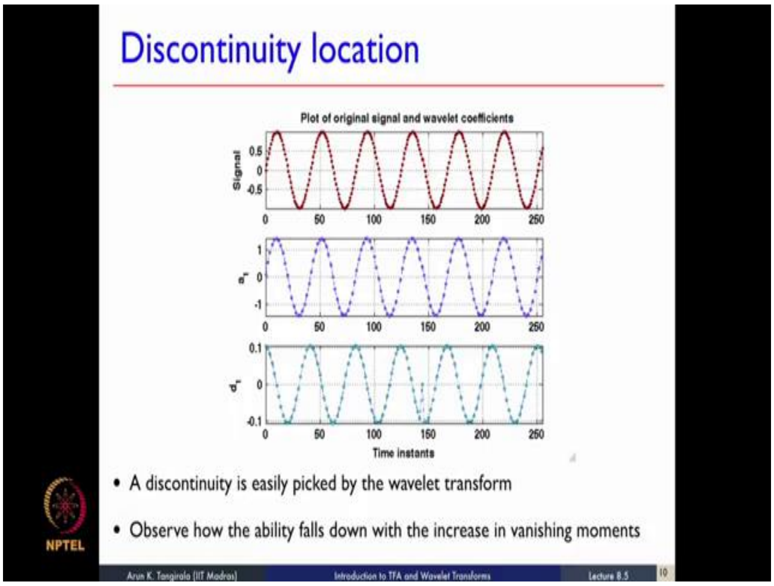
Again applied globally, but a soft thresholding is used and so on. And you can understand what each of this is We have not discussed all the different thresholding methods, such as MDL or the multi sure and so on. But, if you read this paper, it will be very clear to you. What each of this method is doing to estimate the threshold? And also there is something called a form thresholding and garrote thresholding.

There are slightly advanced versions of the soft and hard thresholding. The garrote thresholding performs an interval thresholding. That is, in the hard and soft thresholding what we are doing is, we are partitioning the coefficients into two spaces, one which are below the threshold, other which is above the threshold. In garrote thresholding, the partitioning is done into three spaces. One which is below the threshold lambda one, other which is in between two thresholds lambda 1 and lambda 2 and the third one, which is above the threshold lambda 2.

So, again that is the more advanced one and more sophisticated. But, more sophisticated means also more headache. You will have to specify two thresholds. Then, it becomes sensitive to the choice of two thresholds and so on. So, the basic idea has been laid out. The rest of the method that you see here, for all the methods that you see in the literatures are just flavors of those methods.

Finally, I just want to conclude with this brief mention of discontinuity detection. There is an example here. In the lecture notes, on discontinuity detection which I would like you to go through very quickly.

(Refer Slide Time: 27:25)



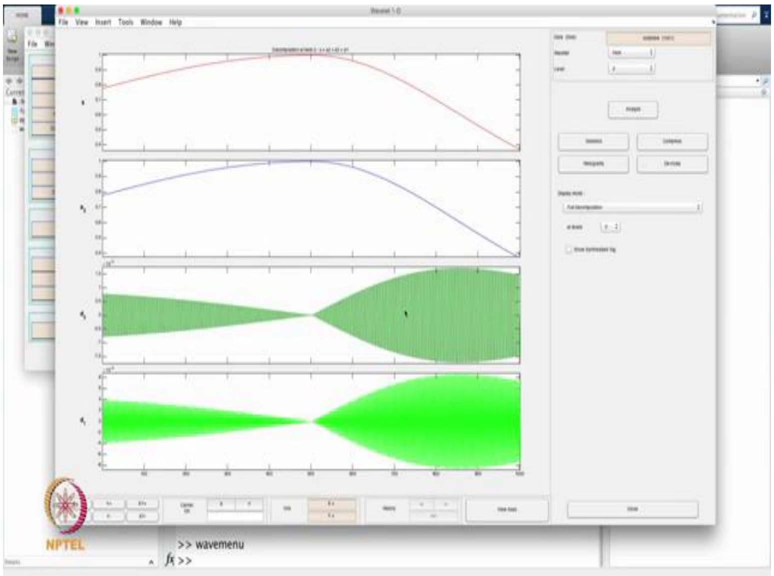
If you look at the example, so here is a sine wave. You do not know where the discontinuity is. The signal is shown on the top. I perform a hard filtering. I am showing you here, the approximation and detail coefficients. You can see in d 1, the discontinuity is nicely picked up, close to 150 here. And then, as I choose a wavelet with higher vanishing movement, so hard wavelet has one vanishing movement.

If I choose d d 2, then its ability to clearly detect the discontinuity falls down. Because, now the spike that I see close to 150 is actually, much smaller in magnitude than what you see with the hard wavelet. So, it is very distinct and pronouns to the hard wavelet. Whereas with d b 2, it is not as pronounced and as I increased the vanishing movements, this spike goes down and fixing the level as it is.

So, what this shows is… The ability of the wavelet to detect the discontinuity in the signal falls down as a number of vanishing movements increases. On the other hand, there is a nice example in the wavelet GUI, which shows you how the discontinuity is detected better by a wavelet with larger vanishing movements than the hard wavelet. Because, discontinuity is not in the signal, but in the second derivative.

And let me just quickly tell you here, what that is. I should shown you earlier, but let me bring up the wavelet only. And if you go under the examples and the basic signals, there is something called a second derivative breakdown. So, one with the second derivative breakdown…

(Refer Slide Time: 29:34)



So, you can see here. This is the second derivative breakdown. If you look at the signal, the signal does not have any discontinuity. The first derivative is also continuous, but the second derivative is discontinuous. For that, you should take the second derivative of the signal. Just take the second difference and you plot, you will see the discontinuity. What is shown here is the approximation and the details, not the coefficients.

Obtain with db 4 at two level decomposition, obtain with db 4. And you can see the discontinuity, in fact we move from construction. This example is also discussed nicely in the book by Misty and Mayor and open hide. So, that is the book that we are being referring to, you can. This also given in the references in mat lab documentation. The discontinuity is in fact at this location, in the second derivative.

Now, if I use a hard wavelet, what do I have? The same level decomposition, what do I see? Now, this is totally different. Why does this happen? Well, this happens because the number of the vanishing movements. Hard wavelet has one vanishing movement. What does it mean? It can approximate polynomials of zeroth order very well.

Now, what is happening here is, the discontinuity is in a second derivative and the approximation order here. That is the error that, hard wavelet makes an approximation polynomial is of a very high. And therefore, you see this oscillating behavior. It is the full detail of, why this oscillating behavior around the discontinuity is given nicely in the book that I have just referred too. So, go back and refer to that.

It is nicely given there, whereas if you increase the vanishing movements, your ability to detect the discontinuities anombegvisely becomes very nice. In fact, if you lower the vanishing movements, you can once again see some oscillating behavior. You can prove this theoretically that, as the number of vanishing movement decreases. And as the discontinuity appears in higher and higher derivative for this signal, you will see an oscillating behavior for the details or for the high frequency components around the discontinuity. I am not going to spend time on that.

But, just to give you feel of what the number of vanishing movements of wavelet, can do to your ability to detect discontinuities in the signals derivatives. So, hopefully these two examples have thrown good light on this fact. With this, we bring a closer to the topic of DWT. The attempt has been to give you, the basics of DWT which will help you Marsh along and learn other versions of DWT, such as wavelet packet transform, maximal over lab DWT.

I have already explained, what is maximal over lab DWT. It only involves discretizing these scales at direct level, but no discretization of the translation. And the wavelet packet transform differs from DWT in the sense that, it also decomposes the detail coefficients at each scale. And that throws doors to a variety of applications, which the

DWT cannot even handle. And it is beyond a scope of the course to discuss the wavelet pack up transform.

So, the effort has been to give you as much as possible, strong fundamentals on DWT. Show you how things are implemented in mat lab. Of course, that is a platform in which we have been showing things to you. What are the practical aspects? And discuss the primary applications of DWT, which is in signal compression, signal estimation and discontinuity detection. So, hopefully we enjoyed the theory of DWT in this unit 8.

And of course, put together with the unit 7. That makes the entire package for you, for wavelets transform. And if you have any question as usual, please feel free to write to us. Good luck and see you in the closing lecture.