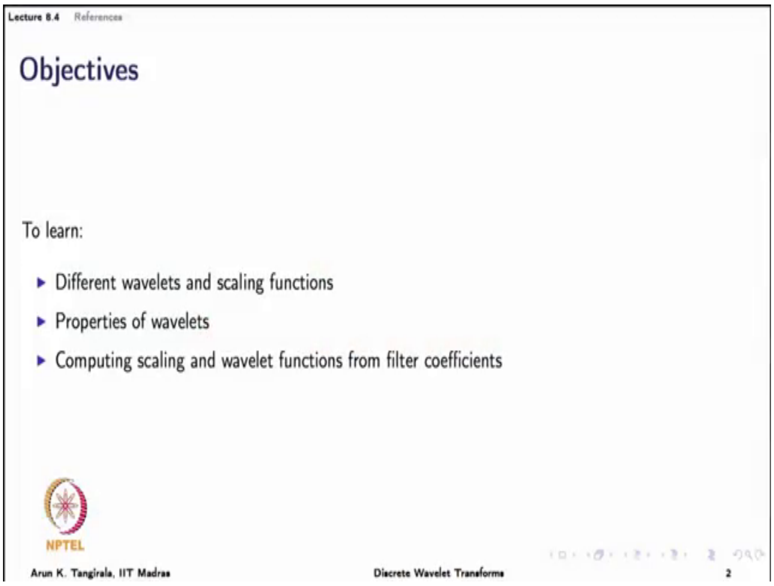
Introduction to Time-Frequency Analysis and Wavelet Transforms

Prof. Arun K. Tangirala
Department of Chemical Engineering
Indian Institute of Technology, Madras

Lecture – 8.4
Wavelets for DWT
Part 1/3

Hello friends, welcome to lecture 8.4 in the unit on DWT. In this lecture we are going to particularly look at different wavelets in scaling functions and study the properties of wavelets that are relevant to DWT. So, all this is in the context of DWT. We have done this exercise earlier for CWT. So, now, it is the turn of DWT.

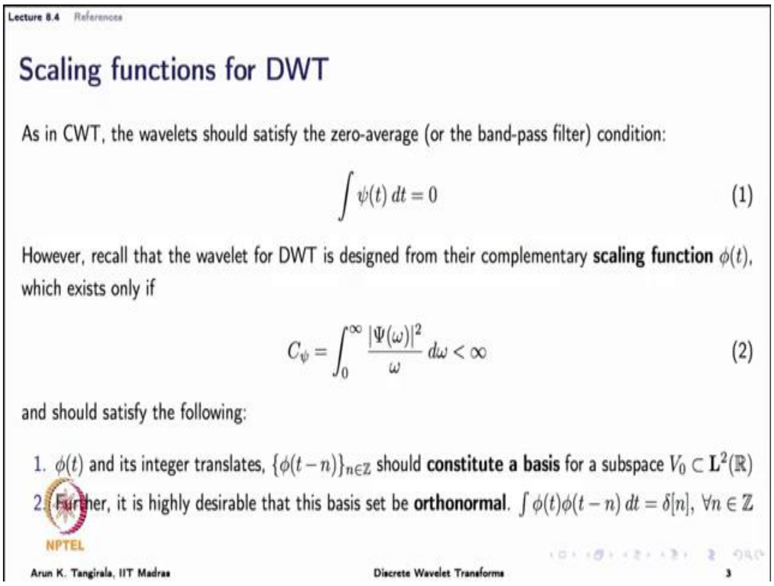
(Refer Slide Time: 00:43)



And also, we will learn how to compute this scaling and wavelet functions from filter coefficients. Remember, we have said that as for as computation of DWT is concerned, both decomposition and reconstruction, one does not need the scaling functions or the wavelets. But still, it is a useful exercise to learn how to derive these functions, how do they look like, what do this basics functions look like, because what we need to do eventually is to select the wavelet for a particular analysis or a scaling function.

And, in selecting a particular family of wavelets we would like to what know features this basics function have. And, unless we look at these functions in time, that is there shapes, we will not able to make much of a decision now. Having said that all the properties of wavelets or scaling functions pretty much can be derived from the filter coefficients, without actually computing the way, this scalings or the wavelet functions themselves, ok.

(Refer Slide Time: 01:49)

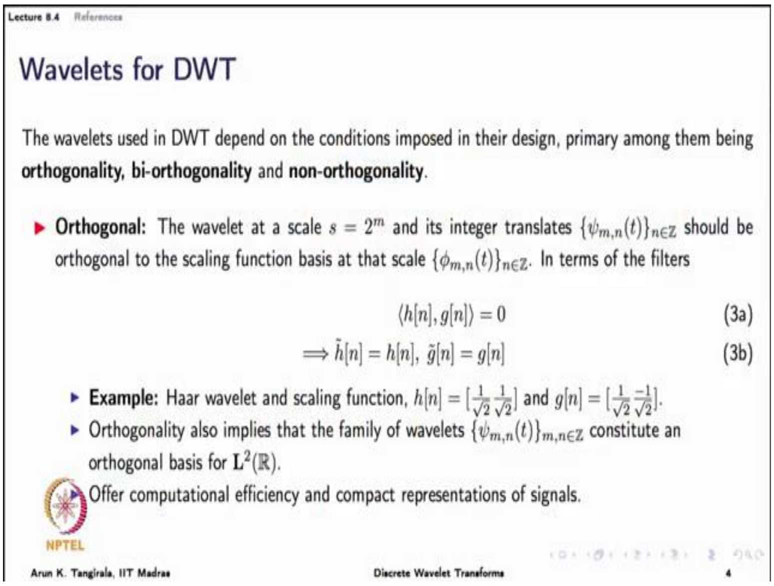


So, let us begin our journey. Just to recall, although I say here, the title of the slide is scaling functions for DWT, it also means that we are going to look at, we are looking at wavelets as well. As in CWT, the wavelets should satisfy the 0 average condition. So, this is just to recap the basic conditions that any wavelet for DWT should satisfy; it should have 0 average.

And, of course, the existence of scaling functions for DWT is very important. And, the prime requirement for the existence of a scaling function is that the wavelet should have a finite admissibility constant. But, additionally, what we require in DWT, is that the scaling function and its integer translates should constitute a basis, and preferably an orthonormal basis for some subspace of finite energy signals in the real space, alright.

So, I hope, it is clear. As for as CWT is concerned a scaling function exist as long as the admissibility constant exist. But, for DWT, the set of admissible scaling functions are those whose interior translate constitute a basis for some subspace of finite energy signals. And, this basis preferably should be orthonormal; why should we seek orthonormal basis functions? Well, as we have learnt in the previous lecture, orthonormality gives us computation efficiency, nice expressions for reconstruction and so on. So, that is the reason essentially, right.

(Refer Slide Time: 03:33)



Now, as far as the different wavelets for DWT are concerned; and let me also help you recall the fact that when we select wavelets in DWT, we normally select this scaling function. And, in fact, we select the low pass filter. We do not really directly select the wavelet necessarily, but one can do that as well. Eventually we use both these scaling functions and the wavelets. So, although, so, it is therefore, justify to begin our discussion to talk about the wavelet for DWT.

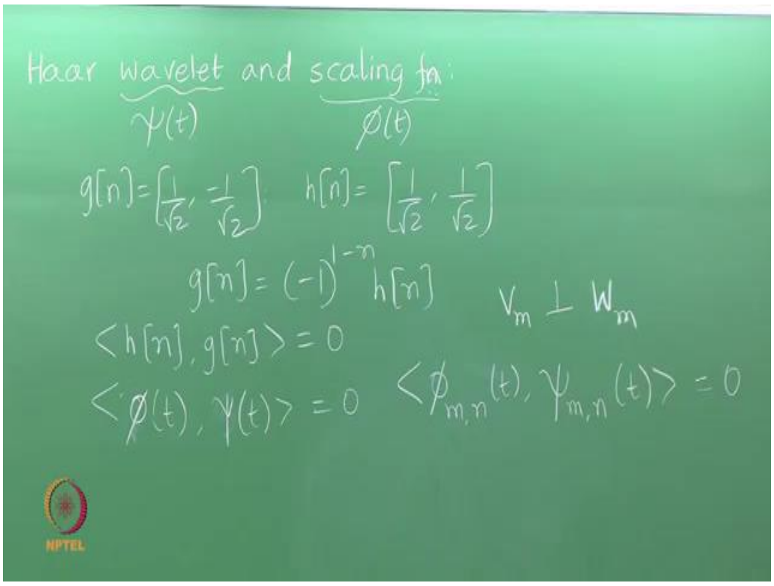
And, we will shortly see that whatever requirements I want on the wavelet, actually translate to the constrains on filter coefficients. We have seen that already in the last lecture, but I am going to explicitly state those constrains here. Now, if you recall the lecture on CWT where we discussed the different wavelets of CWT, we said there are many types of wavelets - there are real value, there are complex value, and then there is a classification based on orthogonality, biorthogonolity, and so on.

In CWT, we restricted our discussion to only real and complex value CWT's, wavelets. Here, when it comes to DWT, we are going to primarily use real value wavelets, but those that are orthogonal or biorthogonal or sometimes non orthogonal. So, let us look at these terms briefly.

We know already what orthogonal wavelets are. What we mean by orthogonal wavelets is, that wavelet at any scale, 2 power m, because we are looking at dyadic's scales, and its integer translates should be orthogonal to this scaling function basis at that scale. So, if I take any scale or any level, m, then I have a set of wavelets at that scale, and their integer translates. They should be orthogonal to the phi t at that scale. So, phi m, n at t. So, this is essentially saying, using the notation that we introduced in the previous lecture, w j subspace should be the orthogonal to the v j subspace; w j is a space of, sorry, w m is the space of details at scale 2 power m, and v m is a space of approximations at level m or it scale to power m; w m should be orthogonal to v m that is what we mean here.

And, once our condition is imposed, as we have seen in the previous lecture, it amounts to saying that wavelets themselves are orthogonal at that scale; that is the translates of wavelets and the wavelets are orthogonal. Now, this condition can be restated in terms of the filters. Essentially, we want the filters to be orthogonal. What we mean, what do we mean by orthogonal filters? Well, the inner product of the low pass and the high pass filters should be 0. So, let us take the example of a Haar wavelet.

(Refer Slide Time: 06:40)

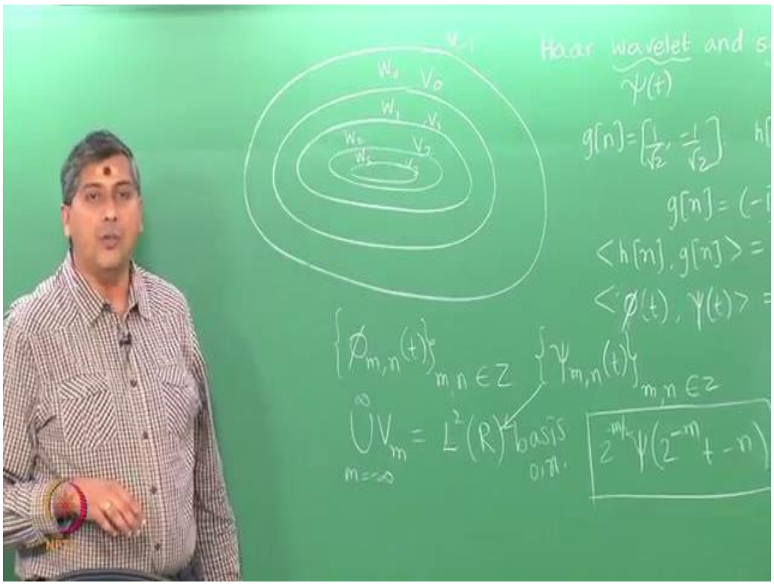


So, we know, for the Haar wavelet or the Haar scaling function, we know that the low pass filter associated with this phi t a c of length 2, and the coefficients are 1 over root 2, 1 over root 2. And, the high pass filter associated with the wavelet is again of the same length, and we know how to derive g from here because the r wavelet is orthogonal to this scaling function at any scale. You can use the result that we gave in the previous lecture, can verify that, g n, is minus 1 raise to 1 minus n, h.

So, you can verify that g is in indeed, this here. And, n, of course, begins from 1 here. So, if I set, n equals 1, then I get, g 1, to be, 1 over root 2; and, if I set, n equals 2, then I get, minus 1 over root 2. So, you can quickly see here that the inner product between these 2 sequences is 0. And, this amounts to saying that the wavelet is orthogonal to the scaling function at that scale, or at any scale in fact. Although we write this at the level, m equals 0, this is also true at any scale for all n; that is at a fixed m, for all m this is true. That is what we mean by orthogonality here. In terms of this subspaces we say that V m is orthogonal to W m; psi m, n spans a subspace W m, and phi m, n spans a subspace V m.

In addition, we have seen in the previous lecture, when orthogonality is satisfied by the wavelets and scaling functions or by the filters, then the synthesis filters are also identical to the decomposition filters. So, that is naturally guaranteed. And, orthogonality essentially means that the family of wavelets, that is now if you look at entire set of wavelets, not just at one scale, but at all scales and their integer translates, put together, it constitutes, this family constitutes an orthogonal basis for L 2 R.

(Refer Slide Time: 10:14)



So, let me again show that to you schematically. So, what we have from MRA is that these scaling functions and the wavelets are spanning a set of embedded or nested spaces. So, if you recall the diagram that we drew earlier, that is in the previous lecture, then if we call is as V 0, then V 0 is spanned by phi t; and then, you have V 1, and then you have V 2, and so on. And, of course, V 0 itself is contained in V minus 1. So, these subscripts are pertaining to the, referring to the value of m essentially.

And, this is V 2, what, difference between V 2 and V 1, that is this space here is W 2 and this is W 1, and so on. So, if I had a V 3 here, then I would have W 3. Now, you can see quickly that V 3, V 2, V 1, V 0, V subscript minus 1, and so on, they are all inclusive. So, the basis functions responsible for these approximation subspaces, they are not independent because one is contained in the other, right.

If I look at the basis functions for all the subspaces, V minus 1, V 0, V 1, V 2, V 3, and so on; in other words, if I look at phi m, n t for all m n in the integer set, this family of scaling functions cannot be a basis for the real numbers basis. Definitely, when you look at all these, the union of V 3, V 2, V 1, V 0, V minus 1, and so on, if you look at the union of V m, as m runs from minus infinity to infinity, that is definitely the space of finite energy signals; that, there is no doubt about. But, what is not true, or what is not true is that this is the basis for L 2 R; this cannot be because one of the prime requirements of basis is that they should be independent.

On the other hand, if you look at, W 3, W 2, W 1, and then W 0, and W minus 1, and so on, they are all exclusive, right. That means, the basis functions responsible for V 2 are in fact, orthogonal, or they are independent of the basis functions in W 3, and W 1, and W 0, and so on. So, what are the basis function for W 3, or W 2, or W 3 and W 1 and so on, they are the wavelets, and that scale, and that interior translates.

Therefore, the basis functions here that span the detail spaces, as m runs from minus infinity to infinity, they can constitute a basis for the space of finite energy real signals. In other words, here if I look at, psi m, n t, again m and n or the entire integer set, they can form a basis; what kind of basis? Orthonormal basis. They are not only orthonormal within subspace W 2, but they are also orthogonal across subspaces, that is what this means, that is a very important point, whereas, for the scaling functions that is not true.

The scaling functions are orthonormal only within a subspace, but if I look at the scaling function in V, for V 1 and V 2, they are not mutually orthogonal. So, when we impose the requirement that the scaling function should be orthogonal or orthonormal, we only impose at a specific scale; we do not impose orthonormality across scales; that is a point.

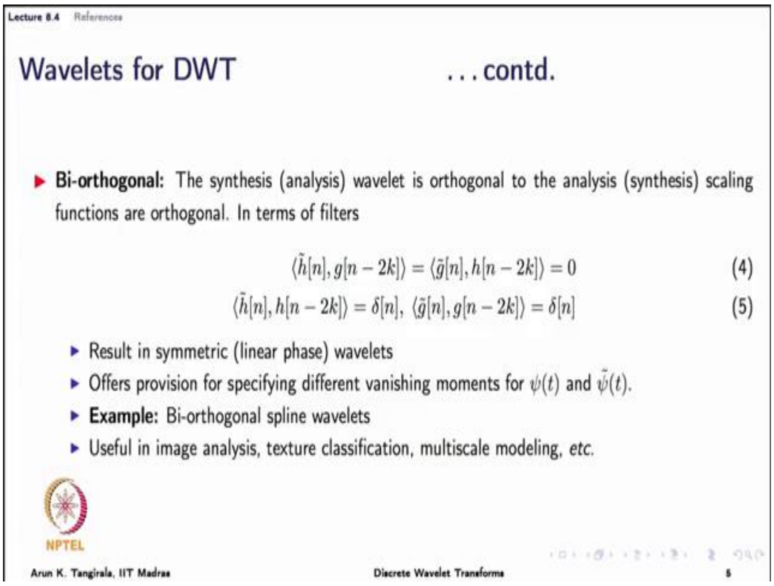
Whereas, with wavelets that is not the case; Vs, they are not only orthonormal at a scale that you, when we say they, the wavelet and its integer translates. That is, at any scale the wavelet family is this, 2 to the minus m by 2 times, 2, times psi 2 to the minus m t minus n). So, for all values of n that is what meant by they, they are orthonormal at a given m and across m as well; that is a beauty of this wavelets.

And likewise, I said here, the union of all the approximation subspaces is L 2 R which is the space of finite and the real signals. The union of all Ws is also L 2 R. That is, I can only look at the approximation spaces or the details spaces, but as far as basis is concerned this do not, does not constitute a basis, whereas, this constitute a basis, in fact, orthonormal basis. So, that makes a big difference, and that is why a lot of discussions begin with wavelets rather than with scaling function.

But, in DWT, because we are interested in approximations; in practice, I am interested in approximations, I want to construct approximations, and the discussion therefore begins from approximation. But, as far as basis is concerned wavelets are the basis functions for the L 2 R.

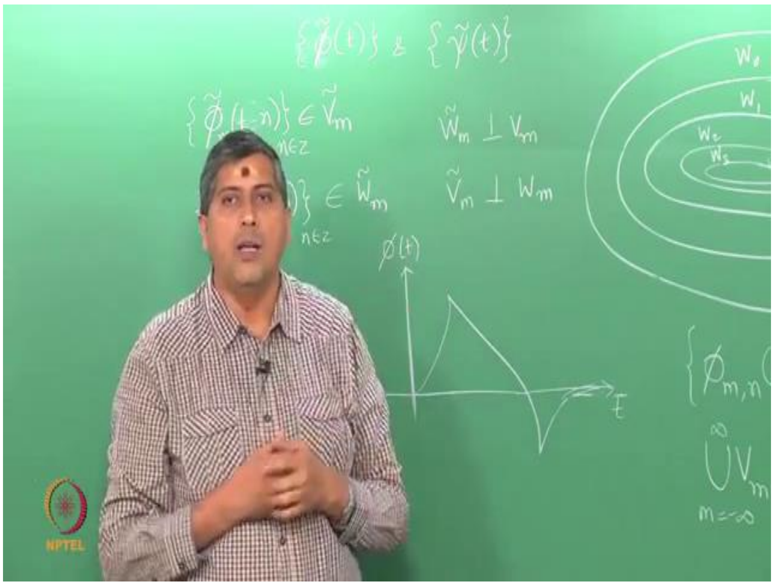
Of course, as we mentioned earlier, when I have orthogonal wavelets, I have the privilege of, luxury of computational efficiency and compact representations of signal; compact meaning, with as few coefficients as possible. Remember, coefficients are the inner products between the signals and the basis functions.

(Refer Slide Time: 16:15)



Moving to biorthogonal wavelets which we did not discuss in detail in previous lecture, but I did mentioned this term when we were talking about synthesis filters, after we discussed the decomposition filters. Remember, we imposed orthogonality at the time of designing the decomposing filters themselves. That is, we said W m is should be perpendicular to V m. But, suppose I do not do that, suppose I say I want a set of wavelets that span the details space, and I already have a scaling, set of scaling functions that span the approximations space at any scale, and I do not impose orthogonality there, but just have some wavelets that satisfy with the condition that they should span the detail subspace; and then look at the synthesis that is when I turn to the reconstruction at that time I require that the synthesis wavelet filter should be orthogonal to the analysis scaling function, alright. So, let me again show what I mean by that.

(Refer Slide Time: 17:50)



Just as we have here approximation and detailed spaces spanned by the scaling functions and wavelets, let us call them decomposition filters, I can also have spaces span by the synthesis scaling functions and synthesis wavelets. We have used this before. We have mentioned that in CWT as well. I can use different synthesis function; synthesis function that is different from the analyzing function. Likewise here, I can use a scaling function for synthesis that is different from the analysis function.

Now, just as we developed this entire theory based on phi t and psi t, I can also have approximation spaces based on phi tiled of t and detail spaces based on psi tilted of t; they are essentially duals. So, you, it is not that only phi t and psi t can generate this spaces, they are also capable of generating this spaces. If you use a notation now, that the subspace span by phi tilde m of t minus n where n is some interior set, let us say this spans V m tilde, and likewise the set of wavelet functions, but the synthesis wavelet functions and there integer translates, span W m tilde, then we arrive at biorthogonal wavelets by requiring that W m tilde b orthogonal to V m, and V m tilde b perpendicular or orthogonal to W m.

So, this is different from what we see in orthogonal wavelets where we require W m and V m to be orthogonal. In this, here, we, it is a biorthogonal. So, you see that W m is not perpendicular to V m, but it is perpendicular to the approximation space generated by the synthesis wavelet functions. Now, at, on the space of it, it sounds a bit weird, as to what does is make, what does this mean?

Well, what this means is that there is a certain symmetricity here, in the reconstruction and the decomposition. And, in fact, this symmetricity here gives rise to what are known as symmetric wavelet filters. You can show that orthogonal filters can hardly be symmetric. You can design symmetric orthogonal filters. What do you mean by symmetry? Well, when I look at the scaling function, for example, the Daubechie scaling function, it looks like this, for example, qualitatively.

So, this is how a Daubechie scaling function would look like, alright, of some order. Now, this Daubechie, wavelet, scaling function is an orthogonal scaling function that is it belongs to the family, it, corresponding to this is a wavelet which is orthogonal wavelet. Daubechie wavelets are not biorthogonal wavelets. Now, you can see that this scaling function is not symmetric. In fact, the associated wavelet is also not symmetric.

So, what happens if there is no symmetricity? Remember, I am using wavelets as filters. And, whenever I use, whenever I perform filtering, whenever I filter a signal, the phase of the signal is altered. And, when I use filters whose impulse responses are not symmetric, although I do not show impulse response here, but phi t is the impulse response of the associated low pass filter. When that is not symmetric then phase distortion occurs.

And, when phase distortion occurs, the sharp changes or the edges in the signal are distorted. And, that is not really good for signal analysis in certain applications. I want therefore, symmetric wavelets and scaling functions with as minimum width as possible because if I have larger and larger width what happens is the time resolution of the wavelet, the mother wave itself will be poor. Remember, ultimately we are looking at time frequency analysis.

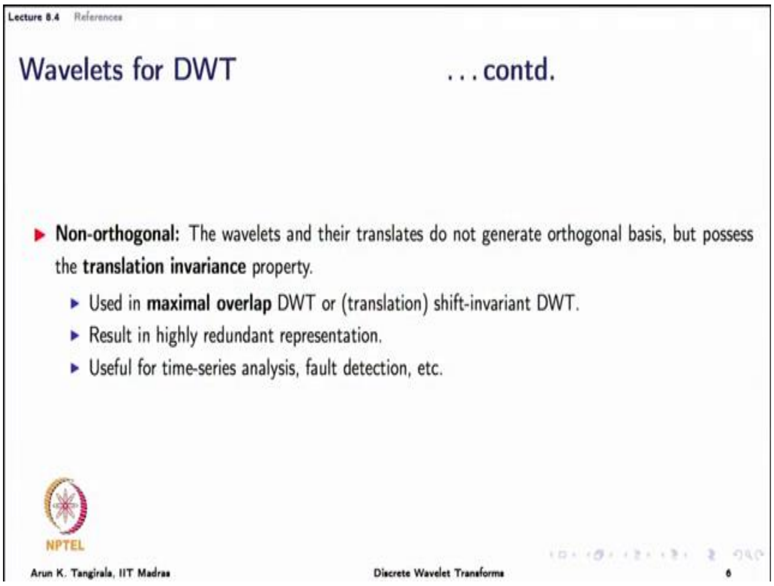
With orthogonal wavelets, to design a near symmetric wavelet you require a much larger support in time. And, as a result, also much larger number of vanishing movements, as usual we will learn later, and therefore, you have to sacrifice on the, that is you have to get a more smooth wavelet and so on.

Whereas, with biorthogonal wavelets, you can nicely design symmetric wavelets. In fact, there is a whole lot of literature. I will direct you to a book, text book which discusses all applications of different wavelets to different processes. And, in that text book you will see that biorthogonal wavelets are very good for texture classification, image analysis, and so on where orthogonal wavelets will generate a lot more coefficients.

Remember, ultimately, one of the main applications of DWT is signal compression. And, in signal compression we are trying, we are seeking wavelets that can represent a given signal in as few coefficients as possible. With biorthogonal wavelets you can achieve this in a much better manner in many applications than orthogonal wavelets; that is because again of the symmetry and the flexibility that you have given. Orthogonality is a very tight requirement, but it is useful in its own way.

One of the examples of biorthogonal wavelets are the biorthogonal spline wavelet. I am not showing you any wavelet shapes here per say; we will do that when we look at the MATLAB session. At this movement you may be wondering why we are not looking at any graphs or any profiles or the functions or wavelets, but I would like to reserve that for the MATLAB session and show you how you yourself can plot this wavelet functions and look at them. I have already shown glimpses of that in CWT.

(Refer Slide Time: 24:15)



Now finally, we have non orthogonal wavelets. This is, these wavelets are also popular for one reason. That reason is when I use orthogonal wavelets which are used widely for signal estimation and signal compression, despite having many advantages one of the key disadvantage of orthogonal, advantages of the, disadvantages of the orthogonal wavelets is that it is not shift invariant or translation invariant. What do we mean by that?

Well, if there is a feature in the signal, let us say, it is a biomedical signal, or a signal coming from vibration missionary, and so on, there are certain features of interest to me, and I would like to keep track of how this features are shifting in time. And, if you recall the way we are computing the dyadic DWT and then the orthogonal DWT, basically what we are doing is we are evaluating CWT at specific scales and also translating at specific translations only, right.

That is we are translating in such a way that we generate non over lapping basis functions. And, that results in minimal representation. But then, what happens is, if there is a feature that is, that are shifted in time, alright, the same feature appears at again at a later time. And, this shift in time for the feature, if that is not commensurate with the shift of the wavelet functions that you are using, then you will lose out on that feature.

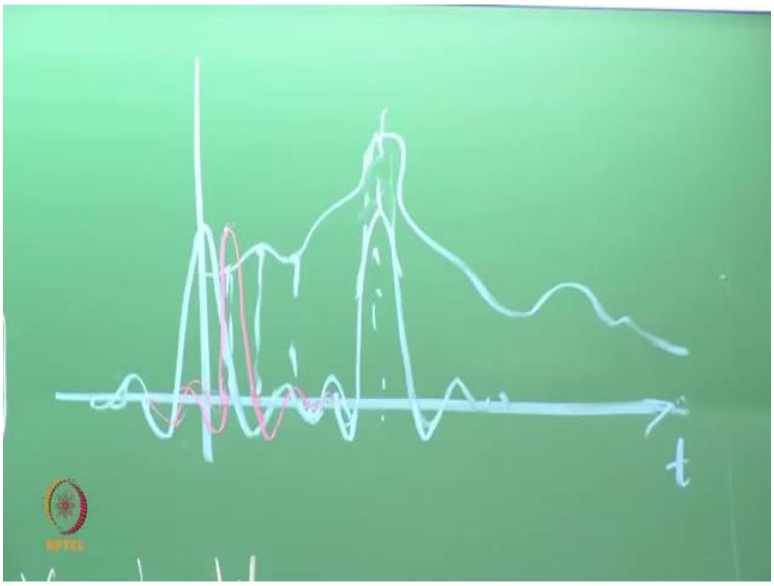
That is, this feature is not, the shift of the feature in the signal, it could be a sudden change or whatever, is not correctly captured by the DWT. That is because you are taking time steps in a particular manner that is much different from the sampling interval of the signal. So, to overcome that property, that disadvantage, what one does is only discretizes the scales and translates, that is chooses a translation parameter equal to the sampling interval. That is chooses to march ahead in time which is equal to the sampling interval.

The tau parameter that we had in CWT is now no longer going to be, n times 2 power m. It going to be simply n where n will be equal to the sampling instant itself which means I will still choose dyadic scales; that is not the reason that I am missing out on the translation invariance. The scales has got nothing to do with it. The reason why I was not able to detect a shift of the feature in time is because I was taking steps proportional to the scale of the wavelets.

But now, I am going to relax that and I am going to let myself, let this wavelet move not proportional to width of the scale, but in steps of sampling interval. There, as a result, if there is feature that has shifted in time, I can easily capture that feature correctly. And, that leads to what is known as shift invariant DWT or maximal overlap DWT. What, why do we have a maximal overlap here?

Because, in time, I am marching ahead one sampling interval at a time. Obviously, the disadvantage of that would be the loss of computational efficiency. I have to compute a lot more coefficients than necessary; as a result we get redundant representation of the signal, and so on. But, this redundancy and the loss of computational efficiency come with an advantage that I will be able to detect shifts in the features.

(Refer Slide Time: 28:35)



Now, clearly, the movement I have wavelets that are not shifting in time proportional to the scale, but rather proportional to the sampling interval, I am going to generate a set of overlapping wavelet. So, let me just show that to you on the graph. Let us say, I have some signal here, and what I am doing in, let us peak a wavelet here which is located at this point; this is some wavelet which is located at the origin at some scale. In DWT what I do is, the next wavelet that would analyze the signal would begin from here and the center would be placed here.

Of course, I am choosing, I am drawing it in very qualitative manner, just, but hope you get the point, right. It is going to be of the same with here, and so on. So, this is n equals 0, n equals 1, at that scale, and so on. This is how I am going to march ahead in time. Whereas, in maximal overlap DWT, the next wavelet, suppose this is the sampling interval here, and so on, the next wavelet is going to be located, is just going to be shifted in time this way, alright. That is how it is going to shift in time.

As you can see, this wavelet here, shifted wavelet, and this wavelet here are highly overlapping. In fact, it is maximal overlapping. Whereas, with the DWT that is the classical DWT, orthogonal DWT, there is not, there is no overlap between this wavelet, the shifted wavelet and this. But, this big shift that I have taken could miss out of feature that would have shifted in time much less than the shift that you have taken at that scale. That is the step that you have taken at that scale.

If there was a feature here, in the signal, somewhere here, and at a later time the same feature appeared here. Because you have taken this step here, in classical DWT, you would miss capturing this shift, and that is the main problem with the classical DWT. On the other hand, because of the lack of overlap you have computational efficiency, you have minimum representation, and so on.

So, I hope now you understand what is the difference between a shift invariant DWT and the classical DWT. A lot of this is discussed in the book by Persual and Walden. And, I will give you a bunch of references in the closing lecture. But, you can just go and read, the book is titled “Wavelets For Time Series Analysis”.