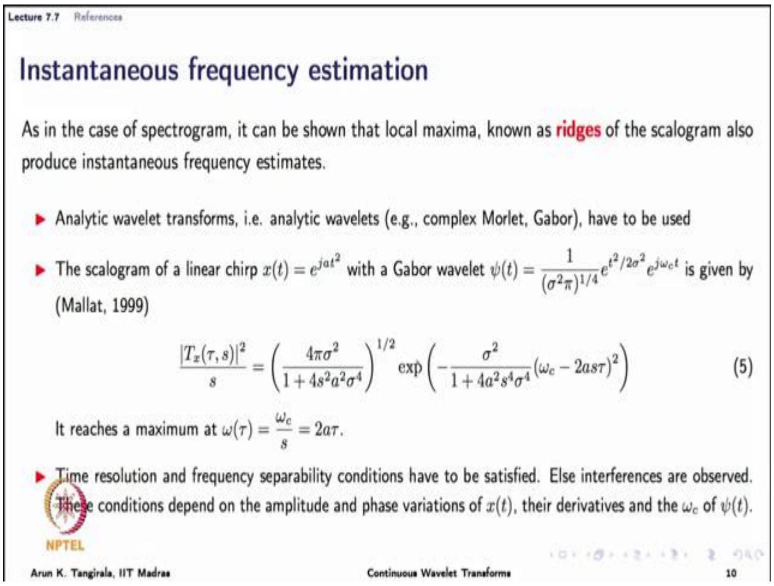
Introduction to Time-Frequency Analysis and Wavelet Transforms
Prof. Arun K. Tangirala
Department of Chemical Engineering
Indian Institute of Technology, Madras

Lecture - 7.7
Application of CWT
Part 2/2

Brings us to the conclusion of how CWT can be applied to singularity deduction. The third application that we want to look at is a estimation of instantaneous frequencies. Now, the idea here is very much similar to what you have seen in spectrogram beside the local maxima, just like we evaluated the local maxima for singularity deduction. We could also use local maxima for computing the instantaneous frequencies. In spectrogram we call this as the Fourier, Windowed Fourier ridges, and in wavelet transform world we call this as wavelet ridges. Essentially what you do is you plot the scalogram and follow the local maxima, and then that is spot the scales or the frequencies at which you see the local maxima and those will give you the instantaneous frequencies instruments. It is important to use analytic wavelet transforms that is you should use analytic or complex Morlet type wavelets to really exploit this property.

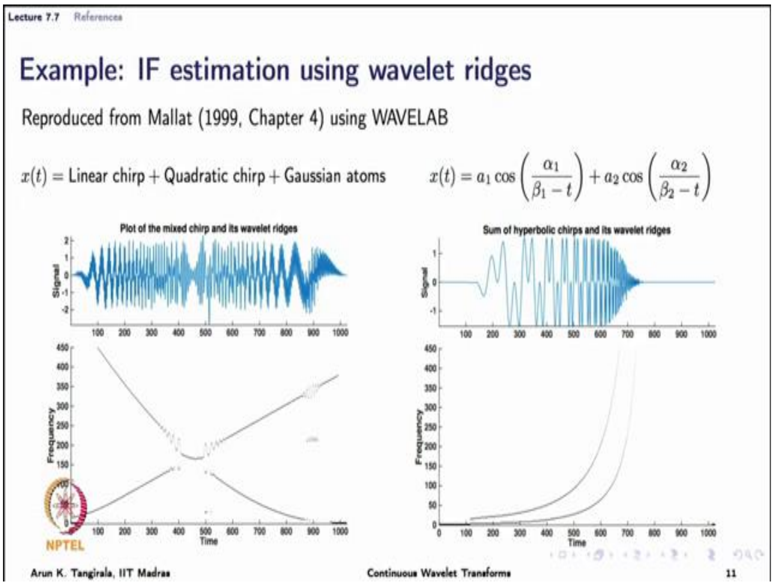
(Refer Slide Time: 01:22)



To give you an example here which I borrowed from Stephane Mallat book, what we do is we perform, we compute this scalogram of linear chirp computer with the Gabor wavelet. Well, Gabor wavelet is essentially a Morlet wavelet and the expression for the normalize scalogram is given here. As you can see from this expression, the normalize scalogram reaches a maximum at omega of tau which is two times a times tau. Now, this 2 a tau is nothing but the derivative of the face of this linear chirp which is nothing but the instantaneous frequency itself. So, therefore, the points of maxima of the normalize scalogram will give me the instantaneous frequencies estimates, these are called ridges.

What is important for your wavelet ridges to really come up with this decent estimate of instantaneous frequencies, it is important that the signal has certain characteristics in itself. That is how the frequency changes with time. It is not that your wavelet ridges will always give you very good estimates of instantaneous frequencies depends on how they are separated in time, and how the amplitude also change, that is what kind of amplitude modulations you have in the signal. So, the mathematical conditions and all of that are given nicely in Mallat's book, but we will not well into that. Let me just show you a couple of examples again borrowed from Stephane Mallat's book.

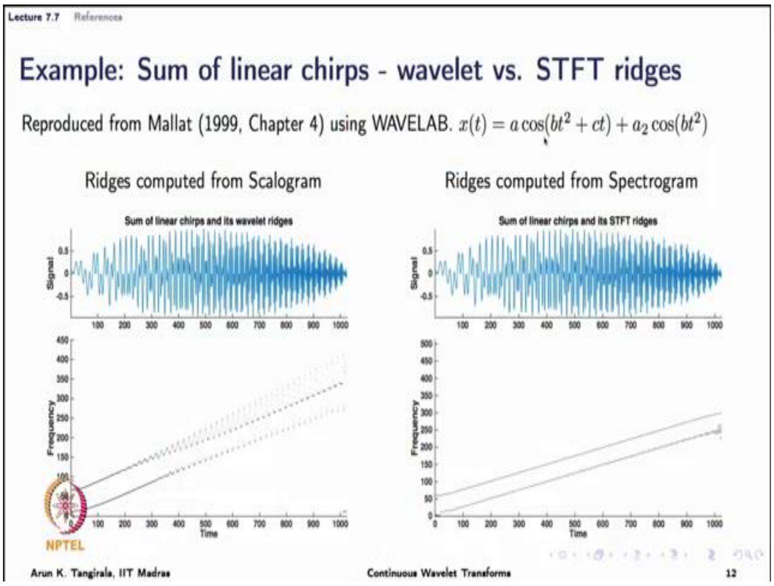
(Refer Slide Time: 03:02)



This again is reproduced from Mallat's book with the help of wavelet. Signal of the left is a mixture of a linear chirp plus a quadratic chirp and couple of Gaussian atoms, and from the figure, you should be able to figure out where these Gaussian atoms are located. First of all, what you see here on the top is a signal itself, and what you see on the bottom are the ridges in computer using wavelet. You can see that ridges have nicely detected the linear and the quadratic varying nature of the frequencies, and you can see some small shadows here, gray shadows here which essentially indicate the presence of the Gaussian atoms, that is where they are located in the signal. In fact, you can generate this signal using wavelet. On the right, we have some of hyperbolic chirps, the expression for the hyperbolic, some of hyperbolic chirps are given on the top. The values of beta and beta, beta 1 and beta 2 are also given in Mallat's Book, but more than the values what is important for us is to see whether the wavelet ridges have correctly come up with the instantaneous frequency estimates.

So, the top here once again is a signal and the bottom here of the ridges. It is clear now that wavelets ridges have come up with very good estimates of the instantaneous frequencies. Recall, if I use standard instantaneous frequency definition for any of the signals, the instantaneous frequency estimates will breakdown. The standard procedure consists of constructing an analytical representation and taking the derivative of the phase. So, that would not work here. What is being done here is we are estimating instantaneous frequencies in indirect manner.

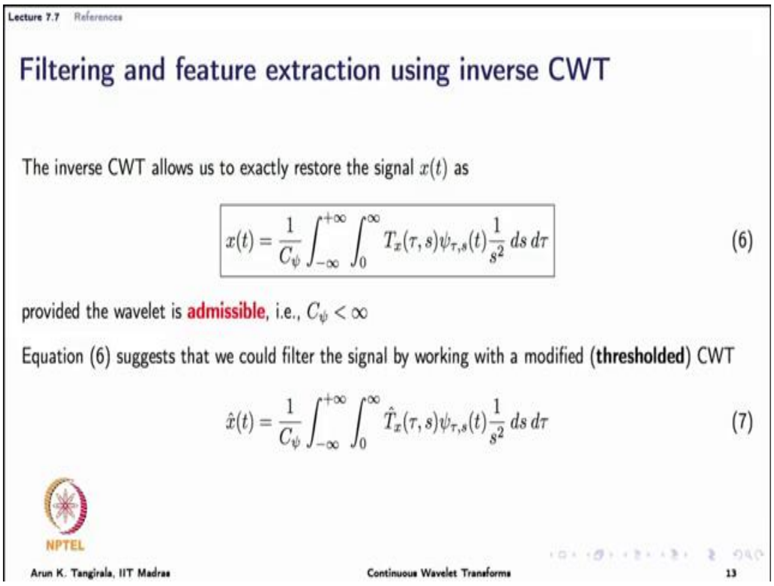
(Refer Slide Time: 04:57)



Let us look at an example where the wavelet ridges really perform a poor job and the STFT ridges perform a very good job. That is a case for the sum of linear chirps again borrowed from Mallat's book in chapter 4, and expression for the some of linear chirps is given on the top, the ridges computed from scalogram are shown on the left, and the one computed from the spectrogram are shown to the right. It is clear from this plot here that the ridges computed from the wavelet scalogram starts showing oscillatory behavior, that is because at this point in time if the nature that is separability that can be achieved by wavelet ridges has broken down. There is a certain mathematical condition that is not fulfilled any longer, the frequencies are to close wavelets to dissolve and amplitude modulations are also not good enough whereas, here the spectrogram is able to resolve this.

Why is there a difference here? It is because when I compute spectrogram, I choose a fixed window with and across entire time frequency (( )), right. Therefore, time frequency resolution or localization is fixed whereas, with scalogram recall that as I move into the high frequency regime, there is more smearing of the energy which means the resolvability of high frequencies is broken down and that is exactly what is happening here. If I had this high frequency in time, initially that is at beginning rather than at later times, then also that would be the same story because it is all about the frequency resolution resolving ability of this particular technique because wavelets use time frequency varying windows that is the band-widths are varying. You have this problem whereas, in spectrogram the time frequency spreads or fixed along with the time scale or time frequency plain, and therefore the resolvability is a same in the entire time frequency. That is the main reason. You would not run into these interferences if these interferences were occurring at low frequencies for example. So, the scalogram would be absolutely fine with that.

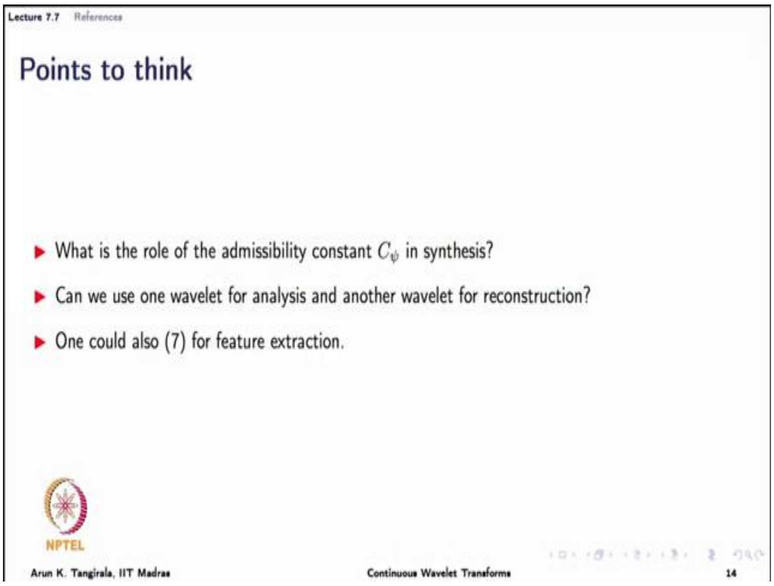
(Refer Slide Time: 07:13)



Now, let us discuss the final application of the continuous wavelet transform that we set out to do in this lecture which is filtering and feature extraction using inverse continuous wavelet transform. Very often there are many features in the signal that I would like to extract and leave aside the remaining features. For this we look at the synthesis equation. You may recall synthesis equation from the first lecture on continuous wavelet transform. T psi is admissibility constant and what this expression essentially suggests is, yes of course this is valid only if the wavelet is admissible. What this expression suggests is that we could reconstruct part of the signal by working with a modified CWT.

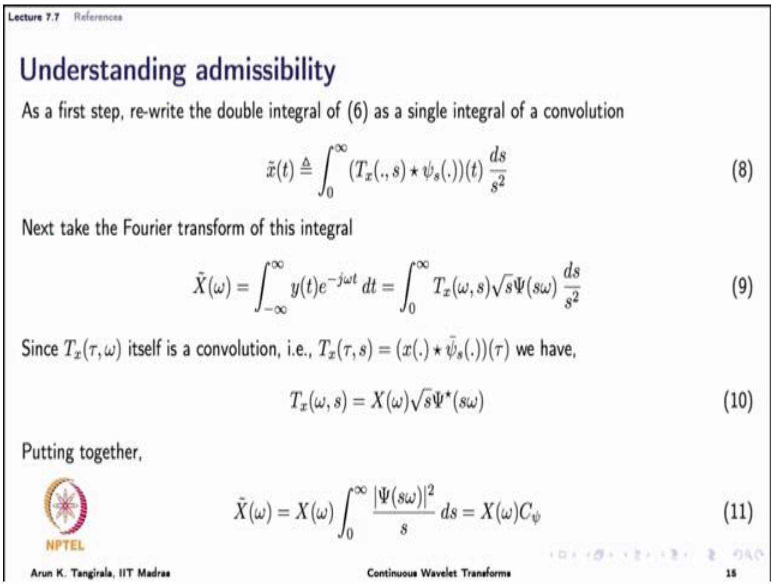
What you mean by modification is thresholding CWT. That is what we could do is, we could take this CWT and zero out a few coefficients like we did in spectrogram or even in periodogram where I showed you how you could filter. Here also, what I could do is I could say that the CWT significant only in certain time scale plane, and I am going to zero out or threshold out the CWT in the remaining region of the time scale plane. Then, reconstruct the part of x of t which is desirable, and this is going to be the basic idea also in DWT. So, therefore, this is kind of a curtain raiser for you for signal estimation using DWT.

(Refer Slide Time: 08:51)



So, let us discuss a few mathematical aspects. Let us understand for example what is the role of admissibility constant in synthesis? This is one of the questions that were also asked in the forums. So, I am going to spend a couple of minutes on explaining why this c psi should appear in this equation and why is it that the wavelet should be admissible for you to recover x of d perfectly. That is the first question we should look at, and second question we should look at is, can we use one wavelet for analysis and another one for synthesis. Can we do that? Well, the answer is yes. Then, how does this reconstruction equation change and thirdly of course, you can use for feature extraction and I will show you an example of that. When I say feature here, it could be oscillatory feature also.

(Refer Slide Time: 09:34)

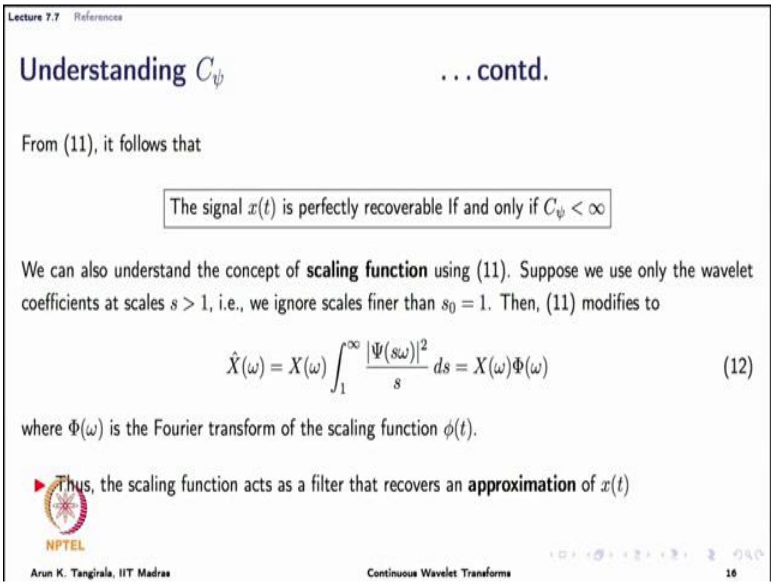


So, let us have a look at the first question. Why is c psi coming into my synthesis equation and why is that wavelet should have a finite c psi? Let us start with the part of reconstruction equation ignoring 1 over c psi. So, what have I done is, I have thrown away 1 over c psi and I am only looking at this double integral, but if you recall here the psi of tau, s is nothing but 1 over route s psi of t minus tau by s which means what is happening is the wavelet transform is being convolved with the wavelet. Therefore, I can rewrite this double integral as an integral of the convolution of t with psi scaled mother wave. So, essentially what we have done is, we have taken one integral and recognize that to be convolution and that is what we have done. The next integral is of course over scales. Now, what we shall and let us denote this with x tilde of t. So, x tilde and x of t only differs by c psi.

Now, let us take the Fourier transform of this integral. So, on the left hand side I have x tilde of omega. This is small error here. This y tilde t should be x tilde of t. Then, by the Fourier transform property, so here I am taking Fourier transform of the integral and we will switch the order. We will say Fourier transform of the integral is integral of the Fourier transform. We can do that under certain mild conditions. Then, the convolution becomes a product in the Fourier domain. Here we are convolving with respect to tau and therefore, Fourier transform is also with respect to tau. We have made use of the Fourier transform properties.

Now, again recall that the continuous wavelet transforms itself is convolution of the signal with reflected version of the wave, right. Therefore, I can write the Fourier transform of the wavelet transform with respect to tau as a product of x of omega times root s psi star of s of omega. Now, when I put together equations 9 and 10, then I have this expression with me, x tilde of omega is x of omega because I substitute for t x of omega with the expression with RHS that I have in 10. So, x of omega falls out of the integral because it is independent of the scale s. The rest of it stays with the integral. I have psi star s of omega, s time of omega time psi of s omega which becomes a magnitude square of psi of s omega. Root s times s I have here as s and that cancels out. One s I have here. In the end I have integral zero to infinity mod psi of s omega square by s ds, and you can quickly recognize that this expression is nothing but my c psi. This integral is c psi. So, x tilde, what we have started off with is the integral ignoring the c psi, and we have shown that if I ignore the c psi, then I will recover x up to this factor c psi. Obviously, to recover perfectly I need to divide x tilde by c psi. That is exactly what we are doing.

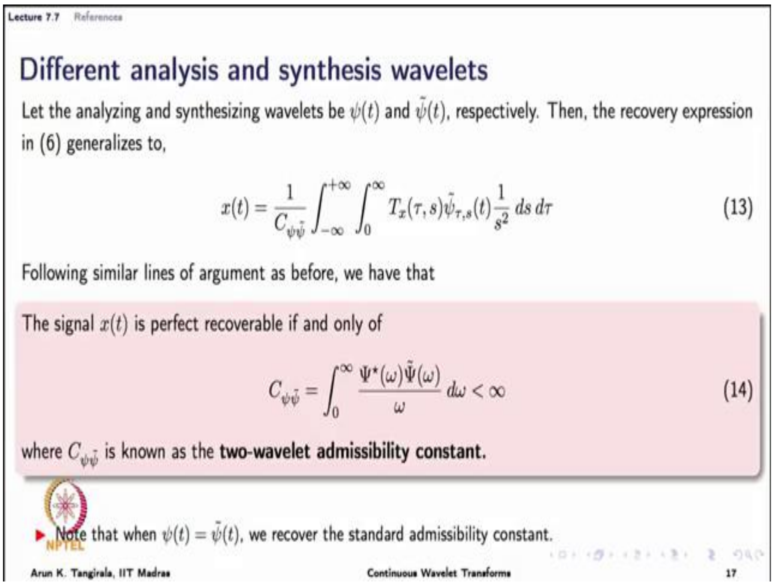
(Refer Slide Time: 12:47)



So, it follows that the signal is perfectly recoverable if and only if c psi less than infinity. It has to be bounded. If c psi is infinity, then you cannot recover it, right. Recover the signal itself. That is the reason why the admissibility constant c psi comes into the synthesis equation. In fact, we can also use this derivation to understand the concept of the scaling function. Suppose we use only the wavelet coefficients at scales s greater than 1 in my reconstruction. So, I go back here into this equation 8 or even in the synthesis equation that I had in equation 6. I only select scales greater than 1 which means I am ignoring all the details below, at scales below 1. Then, what happens to this integral? Well, going by the same procedure we can show that then the x hat of omega. Let us call that as x hat of omega because x tilde of omega looks at is recovered using all scales. X hat of omega is recovered with the wavelet coefficient at all scales greater than 1. Then, x hat of omega is x of omega times is integral. If you recall from the concept of from the lecture on scaling function that we had, this integral here is nothing but your scaling function in the Fourier domain. So, x hat of omega differs from x by this factor phi of omega. So, when I use that wavelet coefficient if had scales greater than 1, then I only recover that part of x which is the x multiply with phi. So, as if I am filtering x with some frequency, with some LTI system whose frequency response functions is phi of omega.

Now, we know already that phi of t access a low cost filters because the phi of omega at omega equals 0 is a non-zero quantity. Therefore, I know phi of omega is nothing but the frequency response function of low cost filter. Therefore, I can say that x hat of omega or x hat of t which is inverse Fourier transform of x hat of omega is nothing but an approximation or low cost filtered version of x of t. So, you see the synthesis equation can be used to understand both the role of admissibility and the scaling function.

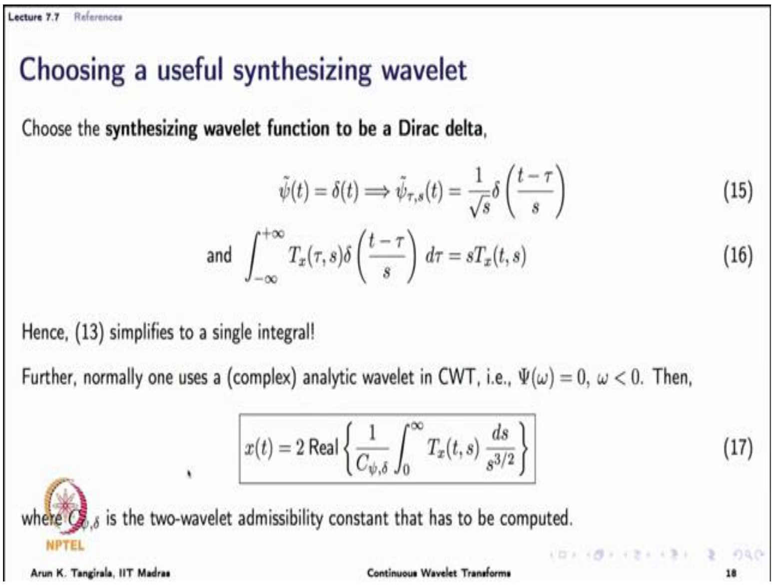
(Refer Slide Time: 15:13)



Now, let us move on to this second question and then we look at an example. Can I use two different wavelets? Why are we even discussing this problem because it turns out that this is the key to coming up with an implementable inverse continuous wavelet transform algorithm. Can I use a different analysis and synthesis wavelet? First of all, why does this idea even come up is, because CWT is a redundant representation of the signal. That means I have computed far more coefficients than actually required. So, I could pick any subset and I can perform any operation on a subset of this coefficient, and I should be able to recover using a different function. That is why I can think of a different analysis and synthesis function. In DWT using orthogonal wavelets, the analysis and synthesis function have to be identical and that we learnt in DWT.

So, let us denote the analyzing wavelet with c psi of t and synthesizing wavelet psi tilde. Then, the recovery expression that we had seen the synthesis equation now generalizes to this integral, where the only difference, there are two differences. In place of psi, I have psi tilde and in place of c psi, I have c psi psi tilde. That means, now I have a generalized admissibility constant which is also known as two wavelet admissibility constant. What is this two wavelet admissibility constant? Well, what this admissibility intuitively this means is that the analysis in wavelet and synthesis in wavelet should be compatible. You cannot really use some two different arbitrary wavelets, one for analysis and one for synthesis. There has to be some correspondence between them and that correspondence or compatibility is measured by this c for c psi psi tilde. In fact, when you said psi tilde equal psi that is when you choose the synthesis function same as analysis function, then you will recover the same condition that we had seen earlier.

(Refer Slide Time: 17:23)

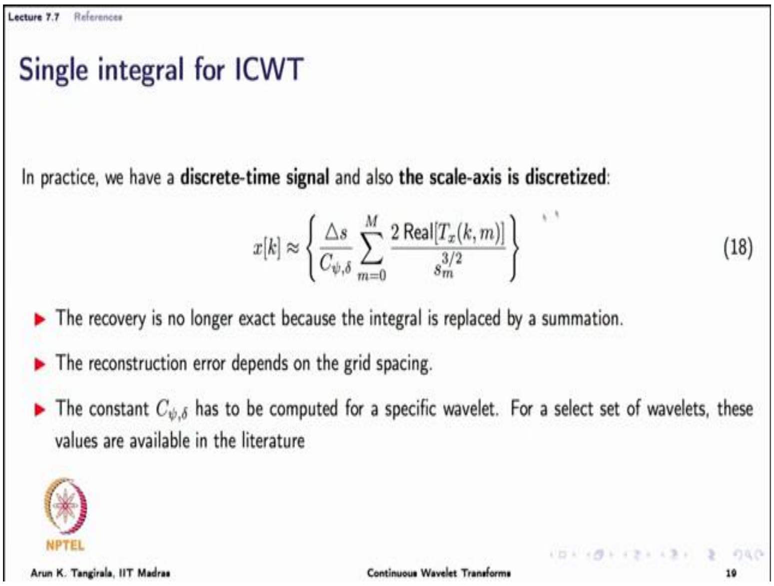


Now, all that is left is to derive the expression that is practically used in implementing the inverse CWT. This is slightly involved topic. Therefore, you may have to pay bit more attention, but there is a very nice trick. If you understand the trick, then the rest of the story is fairly straight forward. What we have here is a double integral and implementing this double integral in practice is a biggest hurdle. We would somehow like to reduce this double integral to single integral. That is what we would like to do, alright. How do we do that? Well, the trick is to use psi tilde as a Dirac. Strictly speaking Dirac is a distribution which has some slight abuse of notation and terminology and some slight giving up of technicality will still call it a Dirac function.

So, I use a synthesizing, Dirac as a synthesizing wavelet. What is an advantage? The advantage is this inner integral that I have or you can say whatever integral we have with respect d tau that reduces to a point, that is if I plug in this expression for psi tilde into this equation 13, then the integral across d tau takes this form. As a result the double integral in 13 simplifies to simple single integral over the scales. That is the basic idea and normally one uses a complex analytic wavelet in CWT. So, what we have learnt is that I can use different analysis and synthesis wavelet, and we use the Dirac function as the synthesizing wavelet because we want to make our life simpler, we want to reduce double integral to single integral and reduce the computation burden. That is the basic essence in inverse CWT. Because we generally use a complex wavelet analytic wavelet for CWT, you can rewrite further that expression that we had here that is the moment you pluck in this simplification in 16 into the equation in 13, you will have a single integral which should have 1 over psi c psi psi tilde times integral 0 to infinity. Sorry, t x of tau, s times 1 over s power 3 by 2 ds. That would be the single integral that we would have.

When you use a complex analytic wavelet, you have to pluck the real version from the wavelet transform. That is how you get this expression. In fact, if you are really curious on how you get the expression from the CWT computed using analytic wavelets, then you can just refer to a very quick three line derivation of this equation of Stephane Mallat book. You can still synthesis even if you do not use complex analytic wavelets. You should remember the fact because normally we use analytic wavelets we have given this expression. Once again here I have c psi delta because now I have chosen psi tilde to be delta. So, see wavelet admissibility constant, that has to be computed and that differs from wavelet to wavelet. In general, we implement at discretize version of this because I cannot compute CWT over a continuum of scales, I would have computed only over a grid.

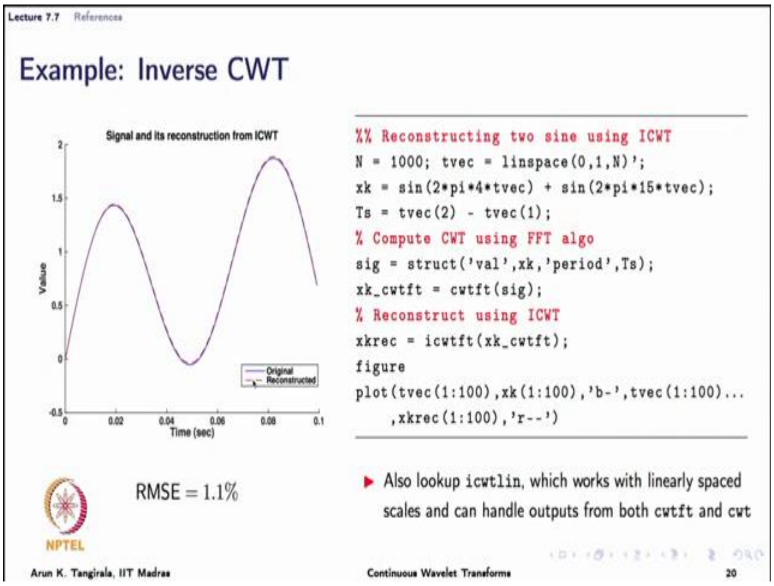
(Refer Slide Time: 21:04)



Now, if I choose to discretize this linearly that is scale axis linearly, then I would get this expression here, right. That is a fairly straight forward expression, but normally one would choose a dyadic grade for these scales in which case the expression in 18 is not the correct one for implementing the discretize version of this. You will have to assume s equals 2 power j and then evaluate ds in terms of j and when I write s equal 2 power j, we vary j linearly and therefore, this integral has to be re-written in terms of j and then discretised. So, that is the basic difference. If I linearly discretize scale, I can directly use an approximate version of 17 as in given in 18, but if I choose a dyadic grade, then I have to rewrite 17 in terms of the linear parameter j and then write the discretize version of that. That is a basic difference and those expressions are given in the literature in the paper by Torrence and Compo and so on. That is what is implemented in Matlab as well.

Now, the only thing that you have to remember is and also observe that we have used and approximation symbol here which means that the recovery is no longer exact. Why because we have replaced integral with a summation. In DWT, this is not going to be the case because the discretization that we choose for scales will allow us to exactly recover x of t. That is the basic difference between performing inverse CWT and inverse DWT. Of course, reconstruction error will depend on the grid spacing and this constant c psi tilde is available in the literature and can be computed for a specific wavelet.

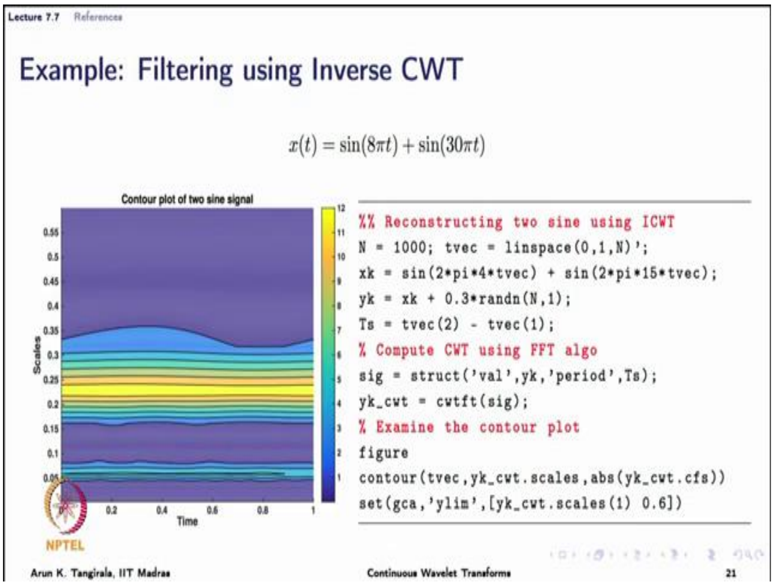
(Refer Slide Time: 22:49)



We turn to 2 examples here. In the first example, all I am trying to show you is simple implementation of how are things in Mat lab. We have a signal sin mixed sin here, and we have generated samples of that. What we are doing here is computing this CWT using FFT algorithm and then computing the inverse. So, what we are doing is, we are not doing anything to the continuous wavelet transform. We are just computing this CWT and then trying to recover the signal. That is all we are doing. We are not performing any operation in CWT, no thresholding.

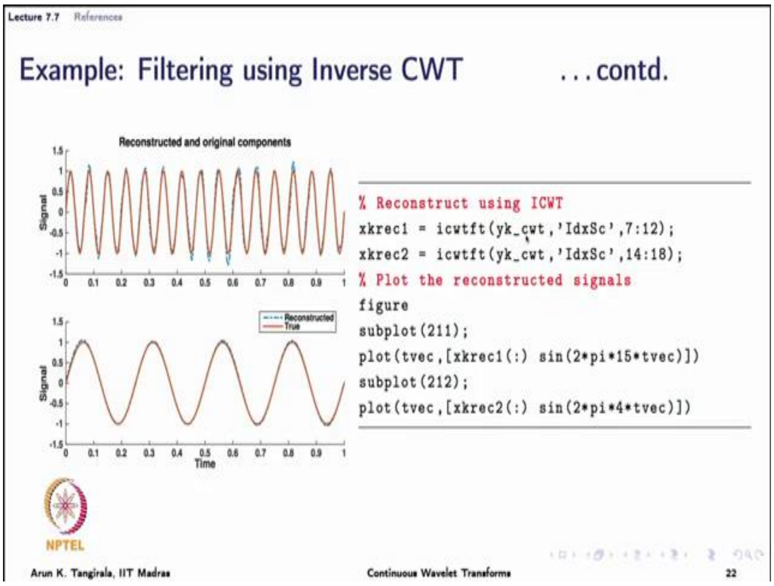
So, I use Icwtft routine Mat lab wavelet tool box. There is an ICWT version that is to be used when you use CWT that is the convolution algorithm to compute or divide the transform. This Icwtft assumes that the argument that you are supplying has been computed from CWT FT. That is important. These are the two complimentary functions and what I am showing you here on the left is the reconstructed version and original version. The original version is shown in blue and the reconstructed one is shown in red. There is some error as we talked about in the reconstruction, and that error is quantified in terms of the route means square error. It turns out to be 1.1 percent that is because we have chosen such grid spacing. If I choose even final grid spacing and I would like you to just explore that. Go and change the grid spacing. When you compute the CWT FT and you will find that you can for finer grid spacing for scales, you can actually get lower RMSE.

(Refer Slide Time: 24:32)



Let us conclude our lecture with example, where we will again take this same example sin 8 bi t plus sin 30 bi t, the same signal that we saw earlier. What I show you here is the contour plot of the CWT. On the y axis, I have scales and on x axis I have time. Clearly it shows me this two time scales that I have in the signal, of course there is smearing of the energy as expected and now, what we would like to know is can I extract the sin 8 bi t which is one part of the signal or sin 30 bi t which is another part of the signal by performing an inverse filtering. What you would technically do is zero out the CWT at those scales that are not of interest to you, and only retain CWT for these scales error of interest to you. That is technically what you could do and that is what you mean by hard thresholding and that is what is achieved by this Icwtft with this IdxSc routine.

(Refer Slide Time: 25:39)



When you use this option in addition to passing the continuous wavelet transform, by the way we have also added some noise here to the original signals, mild amount of noise. So, we may be in the process of filtering. We will also get rid of some noise. So, here I have the component reconstructed of the high frequency part and high frequency corresponds to low scales. So, what I have done is I have looked at this plot and decided that I want to construct the high frequency one which is corresponding to the scales here and I pick the indices of those scales. How do I pick the indices of the scales? From the plot I read what scales are of interest to me and y k underscore cwt has is a structure which contains scales field. So, I got is a vector. I pick the indices that correspond to the scales of interest to me here for the high frequency, and those turn out to be 7 to 12. Those are the vect induces. That is all.

So, that gets me the high frequency component. Likewise I pick the low frequency component. That means, I have to look at higher scales. So, I pick this set of scales for reconstructing. As you can see our reconstruction is really very good. Of course, these are all academic examples, but this actually shows you how you can extract features from cwt. There required few questions on the forum and sent personally to me on how you could extract features from cwt. This is how you could do it. You compute cwt and then perform certain operations on the cwt or if you want the part of x of t that exists over the entire time, but only over select set of scales, then you could use this option, but on the other hand if I only wanted that part of x of t. Let us say between 0.4 and 0.6 in this scale region, then what I would do is I would really zero out the continuous wavelet transform coefficients outside this band of time scale region, and then pass that to Icwtft and that could reconstruct that part of features. That is essentially a procedure for you.

(Refer Slide Time: 28:24)

So, that is it. It was a long lecture, but hopefully a lot of interesting applications for you. You can take this application and see whether they relate to the kind of work that you been doing or plan to do. In doing so not only refer to this lecture notes, but more importantly refer to some of these excellent books by Misiti et al and Torrence, paper by Torrence and Compo and of course Mallat's book, and gratefully acknowledge the software package by Grinsted and Moore, and of course the wavelet tool boxes which is not listed here. I will put that up in the reference list. And I just want to conclude that we have shown is application of wavelets to univariates signal analysis. There are number of applications of wavelets to bivariate and multi-variate signals.

In fact, if you read the top reference, it reads across wavelet and wave coherence, that is essentially looking at wavelet analysis of two signals. So, cross wavelet analysis and there are these routines available in the free package here as well as of course in the Matlab wavelet toolbox that allow you to perform wavelet analysis of bivariate or multi-variate signals. But because this course is fairly restricted and confined to univariate analysis, we have only discussed the applications of CWT to univariate signals. So, that is it so that brings us to the conclusion of CWT. In the next lecture that is in 8.1, we set out on learning what is a DWT, and how does one computed, and so on. So, see you in the next lecture. Bye.