**Biomedical Ultrasound: Fundamentals of Imaging and Micromachined Transducers**

**Prof. Karla P. Mercado-Shekhar, Prof. Himanshu Shekhar, Prof. Hardik Jeetendra Pandya**

**IIT Gandhinagar, IISc Bangalore**

**Lecture: 33**

**Beamforming and Signal Processing**

Hello and welcome to today's lecture. Today we will take a final look at some of the topics related to beamforming and signal processing. So we have taken a brief look at beamforming. You know, beamforming is a spatial filtering approach. You maybe familiar with filtering in general in the context of signals. You may have also heard of things like a high pass filter, low pass filter, notch filter, et cetera, These are frequency filters. They take certain frequencies and reject them. They may be selective in terms of certain frequencies.

Similarly, when we talk about spatial filtering, we essentially enhance signals from certain regions in space and we may suppress signals from certain regions in space. Now, beamforming is used a lot in variety of applications, such as in RF, in audio signal processing, etc. For example, you may have a room where you have a speaker who is talking, and you may want the signal coming from the speaker to be amplified, but signals coming from sideways regions, where other people are talking not to be amplified. So, this beamforming is a very interesting and research active field. And here we'll be discussing it in the context of ultrasound imaging.

Beamforming that we are discussing has two aspects. One is the transmit beamforming and the other is receive beamforming. And if you recall, we discussed DAS or delay and sum beamforming. This is actually the simplest beamforming approach which relies on a constant assumed speed of sound in soft tissue. And it was earlier employed using hardware circuitries. Essentially, you need delays, you need summers, and that is where hardware was being used. Nowadays, it's being implemented digitally.

Beamforming → spatial filtering

filtering → HPF, LPF, NF

Spatial filtering → enhance signal from certain regions in space

Suppress signals from certain regions.

→ Tx , Rx

DAS – Delay & Sum → Employed using hardware ckts.

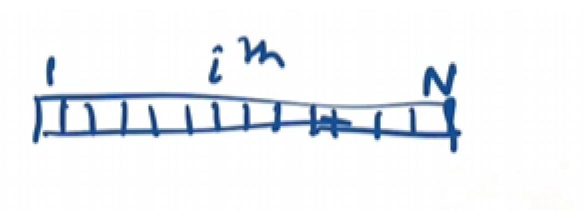
→ Digital implementati

The expression of delay and sum signal in a single line will be the sum of signals Si(t), where Si(t) represents the appropriately delayed signal received by the $i^{th}$ element of the transducer.

$$y_{DAS}(t) = \sum_{i=1}^{N} S_i(t)$$

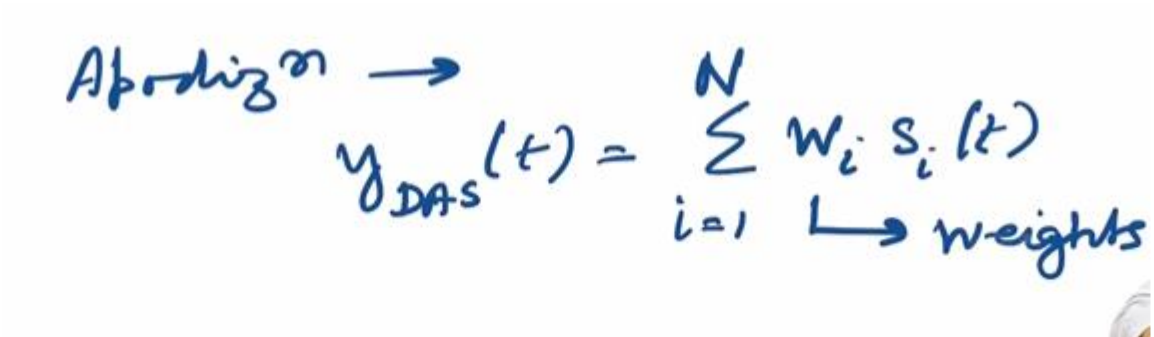
$S_i(t)$ → appropriately delayed signal recieved by the ith element of the transducer

So essentially if this is your transducer and you have several elements and I am talking about the $i^{th}$ element. And i will vary from 1 to N, where N is the total number of elements, which may be something like 64 or 32 or 128 or 256, depending on what type of transducer you use.

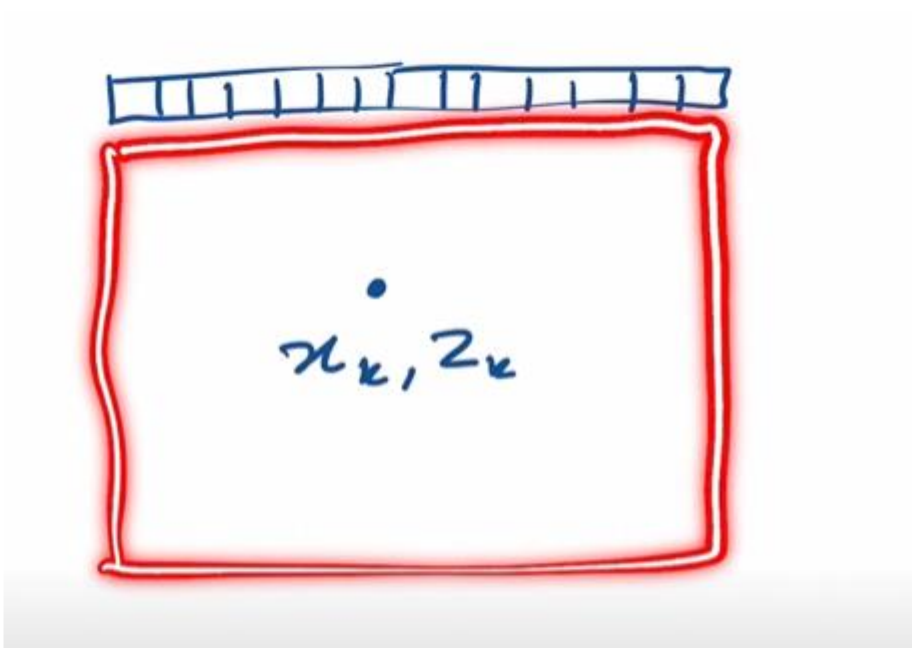


So the signal of the ith element will be appropriately delayed and summed in DAS. When you are firing the signal, you may be firing using different techniques. For example, you may be doing synthetic aperture imaging, you may be doing plane wave imaging, you may be doing focus transmits, and we'll discuss that later. But often, we also employ what is called as apodization.

In apodization, we employ weights. So then your $y_{DAS}$ would be given by a summation which includes these weights. These weights essentially mean, what is the strength of the signal that you are giving either on transmit or the receive signal.

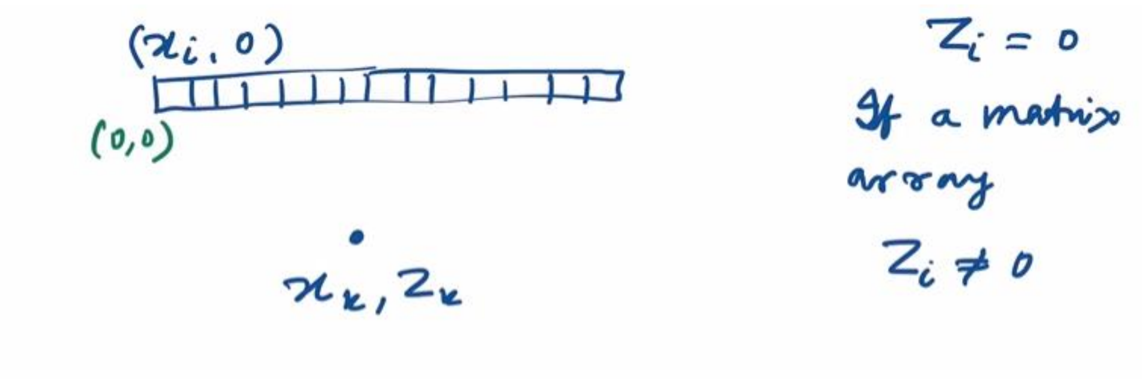



So now consider a transducer array looks like this and any point is given by ($x_k$ $z_k$). So we can assume a grid and the grid's depth will be determined by the depth till which we are imaging. The width will be determined by the array aperture etc. And based on that, in that grid we choose a particular location that may denote a particular pixel. So this coordinate is given by $x_k$, $z_k$

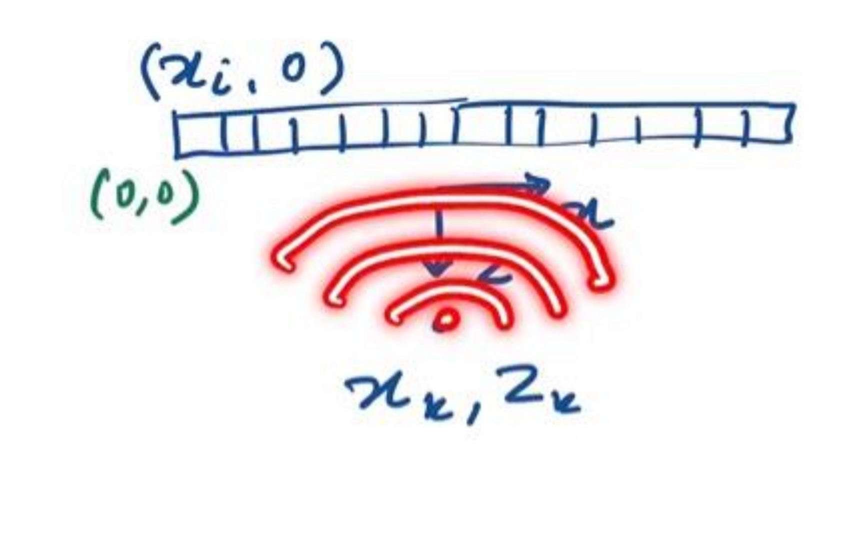



The transducer coordinate is given by $x_i$. and you can also say $x_i$ ,$z_i$. But since this is a linear array transducer, $z_i$ =0. If it is a matrix array, then $z_i \neq 0$. But in this case, $z_i$ =0.

Any particular element in the transducer can be represented by $x_i$ if I consider the left most element as origin which is (0,0).




Now since we know the pitch of the transducer, we know the element to element distance. So I choose the $i^{th}$ element ,which will be at ($x_i$, 0). if a signal is being received from this point marked in red, at what time delay will it be received?



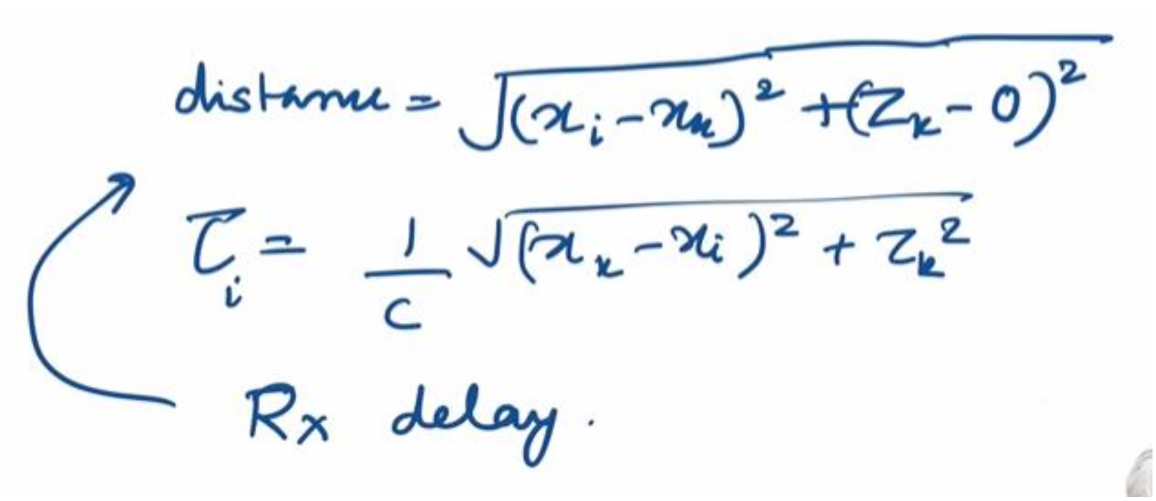

So let's calculate the distance first. If I want to calculate the distance from any point on this array, which is located at xi, then I can use the distance formula from coordinate geometry. So distance will be given by

$$Diatance = \sqrt{(x_i - x_k)^2 + (0 - z_k)^2}$$

$$time\ delay, \tau = \frac{1}{c}\sqrt{(x_i - x_k)^2 + (z_k)^2}$$

This is the delay in terms of how long the signal will take to arrive at the particular element. And as you see $x_k$ and $z_k$ are fixed, because right now this is the pixel that we are evaluating. After we are done with this pixel, we will go to another pixel and so on and so forth. So basically, k will vary across the grid.

However, if you see, $x_i$ location is also going to change. So accordingly, you can calculate delays $\tau_i$ s for the location $x_i$. Now, the total delay, this is the received delay.

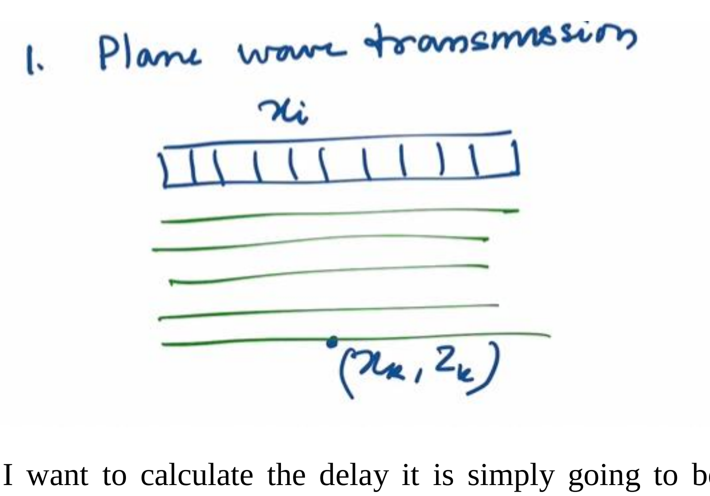



Now, the total delay is going to be

$$Total\ delay = Tx_{delay} + Rx_{delay}$$

And it turns out that the received delay can be calculated just like I mentioned earlier. But the transmit delay actually depends on type of transmission.

So first let's consider plane wave transmission. Here we have some elements and $x_i$ is at any particular location of an element and you have a pixel at a certain location ($x_k$, $z_k$). Now it turns out as this is a plane wave, the entire wave front will arrive at the point at the same time. If you recall in a plane wave the wave front represents a planar surface. So the entire wave front will arrive at the same time.

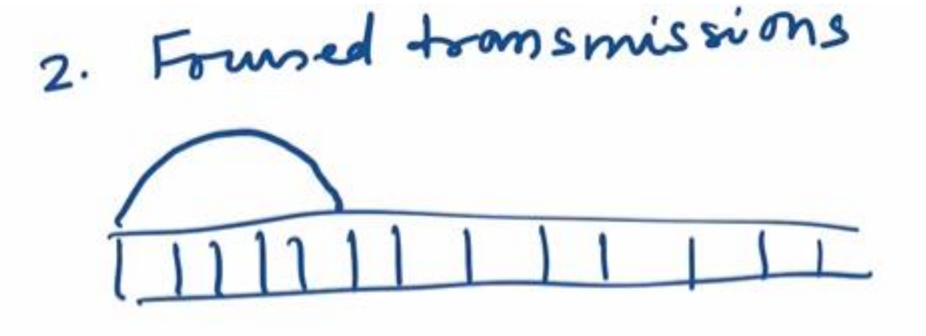



And, actually if I want to calculate the delay it is simply going to be the distance $z_k$ divided by speed of sound, c. This is because it is like a direct transmission and the wave front is a plane.

For tilted plane wave transmission,

$$\tau_i(k) = \frac{1}{c}(z_k cos\theta + x_k sin\theta)$$

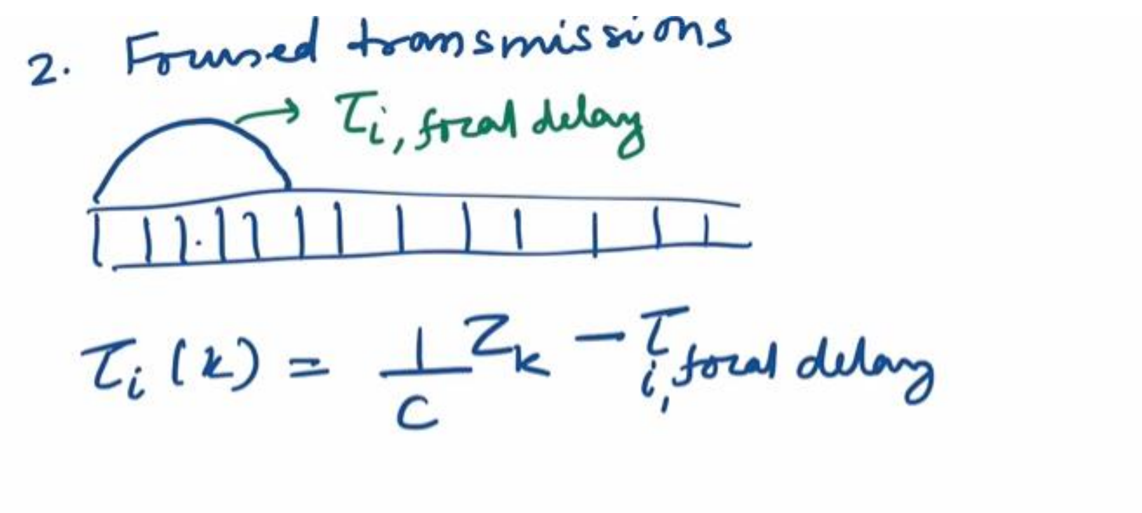
Let's consider focus transmission now. In focused transmission, you have what is called a sub aperture. You may be using some of the elements. Let's say these elements you will be applying a delay and an apodization and you will be firing.




So in this focus transmission what happens is

$$\tau_i(k) = \frac{1}{c} z_k - \tau_{focal\ delay}$$

Essentially, to focus the beam, you have applied the delay and so each element will have a corresponding focal delay which is represented by this delay profile. For example, here the least delay may be applied to outer elements of the subgroup, and the highest delay may be applied to the central element of the subgroup.




Once focal delay is subtracted, It becomes like a plane wave. So I hope you agree with this. Once I subtract the focal delay, the wavefront will become planar.

So now if I have a tilted wave, then what happens?

$$\tau_i(k) = \frac{1}{c}(z_k cos\theta + x_k sin\theta) - \tau_{ifocal\ delay}$$

I hope this point is clear on how you can calculate the transmission delays

Next type of transmission is synthetic aperture imaging. In synthetic aperture imaging, each element is individually firing. So there are essentially no delays in transmission or essentially no focal delays to worry about. So if I have a point at $x_k$ and $z_k$, then I can calculate the delays simply based on

$$\tau_i(k) = \frac{1}{c}\sqrt{(x_i - x_k)^2 + (z_k)^2}$$

And like I said, we need to sum up the transmit delay as well as receive delay to be able to get the total delays. So once I have the transmit delay and the receive delay, that is my total delay.

So my process summary is, First you estimate the delay and then you create $S_1'(t)$ which is the appropriately delayed signal, and let's say my signal is $S_1$. So we have ,

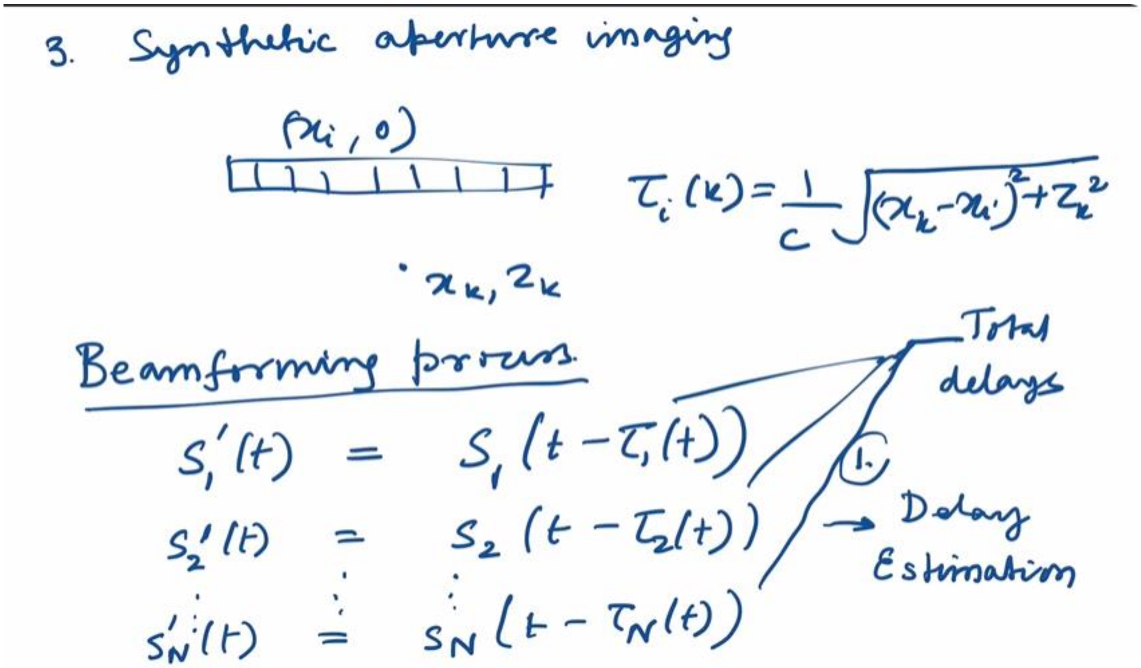
$$S_1'(t) = S_1(t - \tau_1(t))$$

$$S_2'(t) = S_2(t - \tau_2(t))$$

……………………….

$$S_n'(t) = S_n(t - \tau_n(t))$$

Here, the $\tau_i$ s are the total delays. And once you put these total delays, it will have the effect of bringing the signals in phase. And when you sum them up, then it will give the maximum possible signal. So first step, like I said, involves, this is what I call delay estimation, which we have done.




The second step will be apodization. The signal from each element is given a weight.For the elements in the transducer, we will be providing a weight. The weighting function is shown in figure below.


Remember that here when I am weighting, it is essentially a gain of the transducer elements in terms of amplitude and this is not the delay profile that I am showing. There might be another delay profile also that is associated. But this is the relative amplitudes that is represented by this profile. This tells me that the center elements has the highest amplitude and the lowest amplitude is for these peripheral elements. If I do this apodization the weight $W_i$ (t) represents,

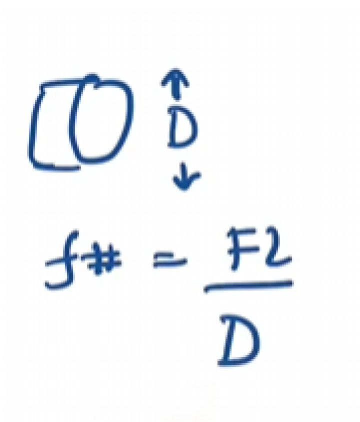
$$W_1(t)\, s_1'(t) + W_2(t)\, s_2'(t) \cdots W_N(t)\, s_N'(t)$$

I will sum up all the way up to $W_N$ (t) $S_n{}'$ (t). This is my signal which is also apodized with proper apodization weight. So if you recall apodization, it literally means foot shaping. You will recall that the word pod, like bipod or something like it denotes foot so foot shaping essentially means footprint shaping. Because the array aperture actually determines its footprint. So you are shaping its footprint by providing different weights to the transducer element and if you recall apodization helps with aperture control. It tells you how many elements are chosen and what are their weights. Their weights essentially is the voltage profile that is given to all these individual elements.

Typically beamforming changes with depth. This beam forming process is being done pixel by pixel. So as we are looking at pixels that come from shallow regions as well as deeper regions, we can choose to change the apodization. So apodization changes with depth. Why is that necessary? First of all, let's think about aperture. I think you must have heard of aperture in the context of camera imaging where it denotes the diameter of the sensor array that is used for photography. Similarly, aperture here, it represents the active elements that are being used for this imaging process. And it is a combination of transmission as well as reception. When you are sending the signal, you may have a different set of elements. When you are receiving, you may have a different set of elements, or a different number of elements. So the effective combination of that is our aperture. So let me define something called focal number or f number. This is also used in imaging, camera imaging etc.

$$Focal\ number\ or\ f\ number\ = depth/aperture\ size$$

Now consider a single element disc transducer like this with a diameter d. For such a transducer we can define f number easily, because this transducer has a fixed focal length. So f number is defined as focal length by aperture diameter which is D.




But here in our case, we can change the focus because we have an electronic array. So we are defining focal number as depth by aperture size so what we do is that we try to keep f number fixed throughout the image as we go through the different pixels and we do beamforming for all those pixels We try to keep f number fixed. So in that case what this tells me is that if I have a pixel that's deeper, means it is located in a deep location. So then as you see the depth is increasing, it means the aperture size should also increase.

Similarly, if I have a pixel that is located a shallow region then the depth is lower so I can use a smaller aperture.

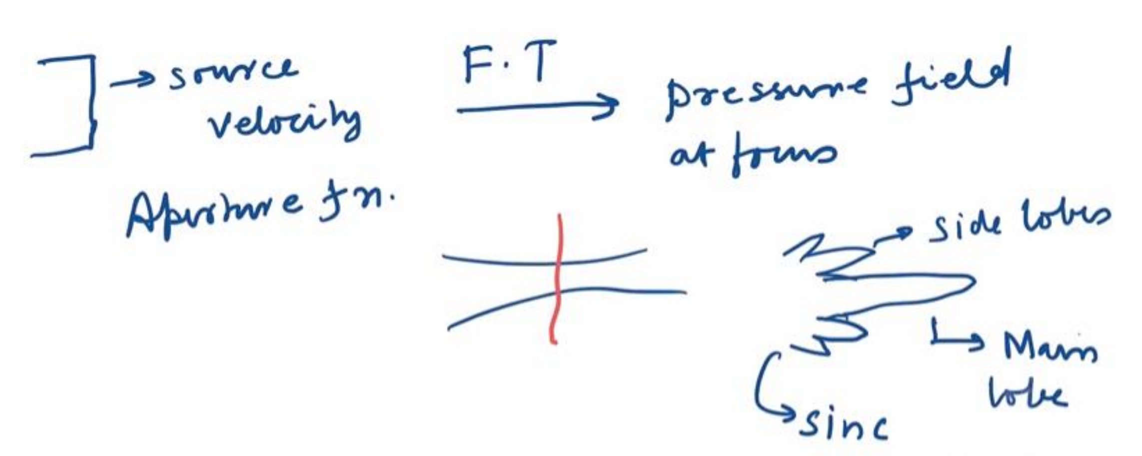
Now it turns out that there is something called dynamic aperture in which you increase the aperture size as you go deeper to maintain same f number. Sometimes it may be difficult because you may have limited number of elements in your aperture. So as you go deeper and deeper it may not be possible to keep increasing the size of the aperture. But to the extent possible what you would do is that you would use more and more elements to fire and to receive as you go deeper in the tissue.

Dynamic aperture

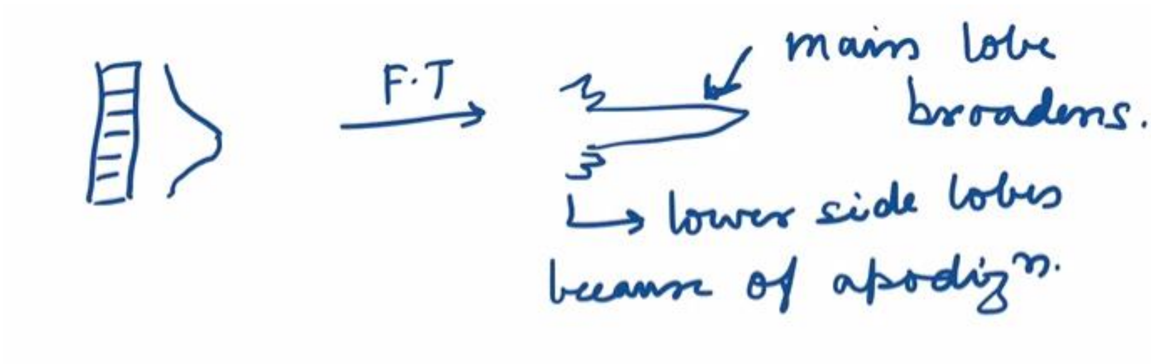
→ ↑ the aperture size as you go deeper to main same focal #.

→ Grow aperture (use more elements) as we visualize deeper targets.

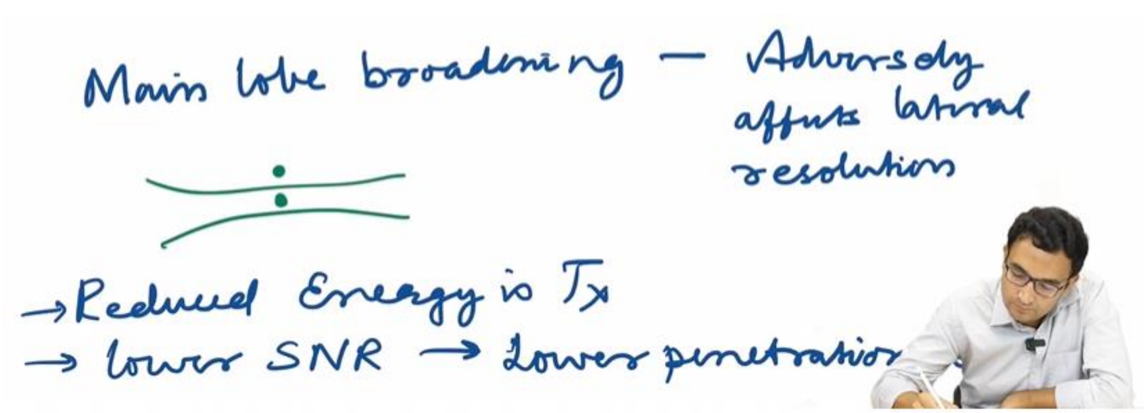
Let us talk a little bit about the advantages and disadvantages of apodization. We looked at this qualitatively the other day as well, where we said that it reduces side lobes. So what are we talking about here? If you remember that if you have source velocity and source velocity will take the same form that will be proportional to the aperture function. So here the aperture function is a rectangular pulse like signal. So if you remember there is a Fourier transform relationship between the source velocity or the aperture function and the pressure field at focus.




So if you take a cross section of the beam, what you get you may get a profile as in the figure below with the main lobe and the side lobe. This profile is going to be related to the Fourier transform of the aperture function or the source velocity function. In this case, the source velocity represents a rect signal and so the profile looks like a sinc signal and sinc signal has its own ripples or side lobes. So we would like to reduce side lobes. Why should we reduce side lobes? If you will recall, side lobes determine our contrast resolution. That is why we would like to reduce the side lobes. If it's a single element transducer, then there is not much that we can do to change the source velocity function or the aperture function. But if we have an array of transducers, then I can apply an apodization like this this which will essentially smoothen the aperture function and when I take a Fourier transform then and if I look at the profile then I get a main lobe and my side lobes will become reduced.




But as it turns out in signal processing, there is always a trade off. To gain something, you have to lose something. So what do we lose here? It turns out the main lobe broadens. So what's the challenge with main lobe broadening? If you recall the lateral resolution, which is the ability to differentiate two scatterers which are within the beam and outside the beam. If my main lobe broadens, then it adversely affects lateral resolution. So it affects lateral resolution, but it improves contrast resolution. The other aspect is if you are providing this kind of a weighting function, then you are telling the peripheral elements to fire with lesser energy because the weight given is less here. So reduced energy is transmitted and reduced energy means lower SNR and lower SNR in turn will mean lower penetration depth.

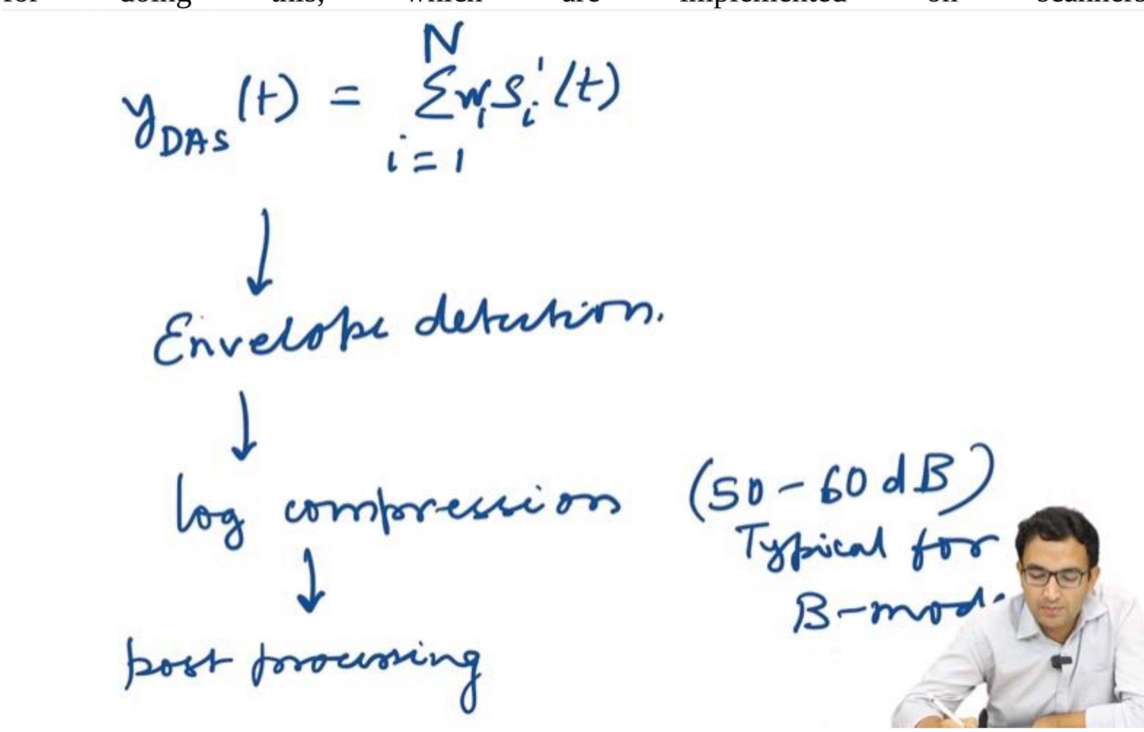



So all these aspects have to be considered. But nonetheless, apodization is used widely because of its ability to improve contrast resolution and it sometimes dehazes the image if you go for apodization.

So, essentially after our beamforming process is complete for each element, we have summed the signal. If you remember the formula where we are getting

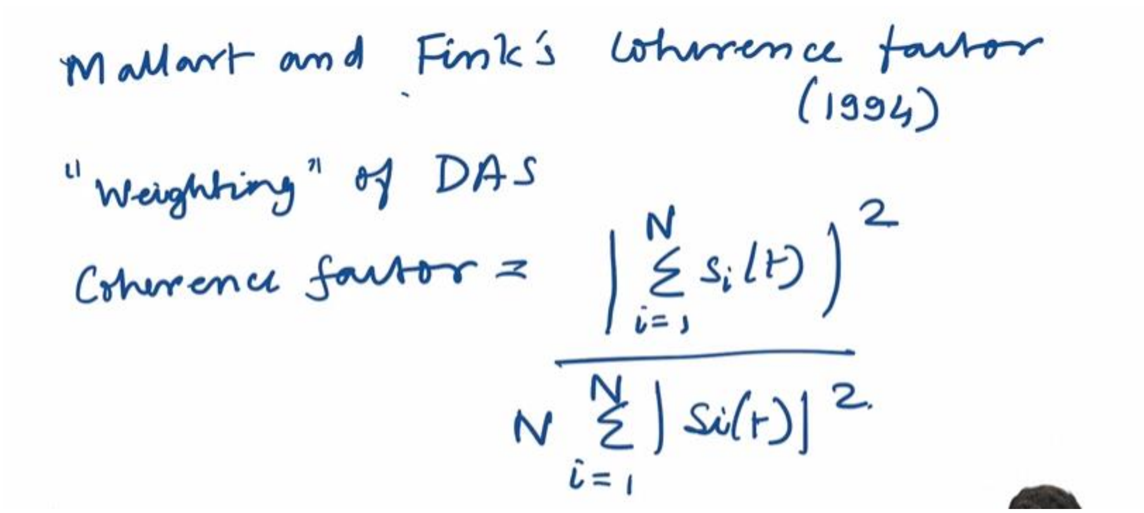
$$y_{DAS}(t) = \sum_{i=1}^{N} W_i \, S_i'$$

Once you have this signal and you have summed it up, the next step would be a process of envelope detection and then you would go for log compression and you would show the image at the appropriate dynamic range. 50 or 60 dB is very common for ultrasound images. Based on the application, it can be higher or lower also. For example, in contrast enhanced imaging, typically when you are using non-linear imaging, it typically uses lower dynamic range because the majority of the dynamic range is occupied by the linear signal which gets cancelled off. So the remaining signal occupies the lower dynamic range. Once you have the image, you can do all kinds of post processing. Depending on the type of image, some scan conversion may be needed, some filtering, denoising, some image sharpening, all those can be done. There are a lot of proprietary algorithms as well for doing this, which are implemented on scanners.




The delay and sum beamforming is the simplest one. In our previous lecture, we just briefly touched upon some other advanced beamforming techniques, such as minimum variance beamforming, robust capon beamforming. We also discussed delay multiply and sum beamforming. Next, I would just like to discuss a very simple trick, which is a way to enhance the performance of delay and sum beamforming.

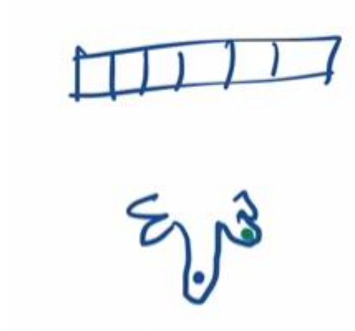
This technique was reported in 1994. It is a weighting of the DAS beamformer. So whatever signal you get from delay and sum, you take it and you weight it with this factor called the coherence factor. It is given like this




So if you look carefully it's a ratio of coherent sum and incoherent sum across the array elements. What is the coherent sum? In coherent sum, we take the phase into account. So here, for example, if we take these $S_i^{'}(t)$, these are the appropriately delayed signals. So if you take these signals, and sum them up, then the phase will matter. There will be constructive and destructive interferences. So once you first sum them and then you take the energy which is proportional to the modulus square, then you will get what is called the coherent sum. But if you first take the absolute value of the signal, then you lose that phase information and then you sum them up. This is called the incoherent sum, which comes in the denominator.

So if you take the ratio of this coherent sum and incoherent sum and use it as a weighting factor, it improves performance. Increased contrast and resolution can be obtained from coherent factor weighting of DAS beamformers. And there are many other factors like that based on coherence properties.

For example, there is phase coherence factor, there is sign coherence factor, and you can certainly look into the literature for more. But nonetheless, the basic principle remains the same. So let's look at it like this. Let's say this is my transducer array, and I have a main lobe, marked as blue point and I also have side lobes of my ultrasound field.



There is another point, labelled as green. Now it turns out that the signals arriving from the main lobe, I am actually putting them in focus, because I am applying delays by taking this blue point's pixel into account and I am bringing all those signals in focus in phase and then I am summing them up. But when it comes to the green point, not all the signals emanating from this location will be in phase. So the maximum coherent sum will be for signals emanating from the main lobe and it will reduce for signals emanating from the side lobe. And hence we get some reduction in the side lobe, and as the side lobe drops, so do the artifacts associated with the side lobe.

So I think this gave you a good information about beamforming and gives you some details in terms of how it is actually accomplished. You can also look into how it is done computationally on systems. But I think we'll stop here. And in the next class, we will do a recap of beamforming and image reconstruction. Thank you.