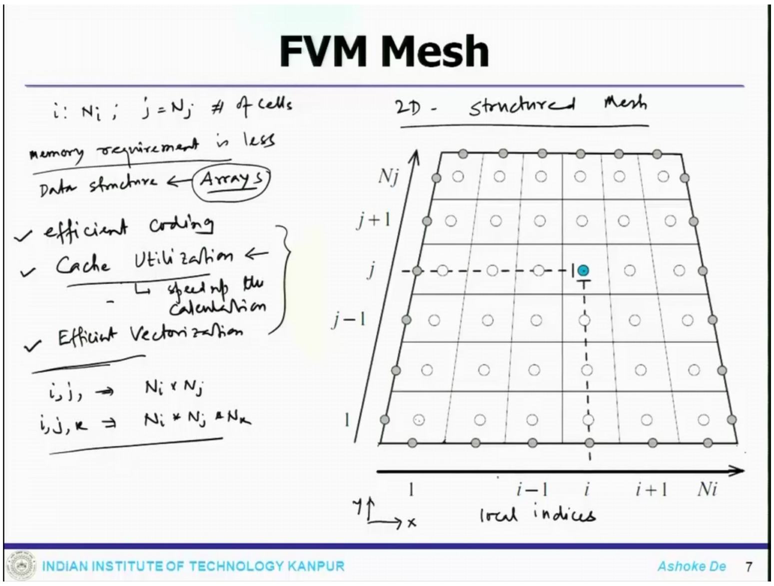
Introduction to Finite Volume Methods-I
Prof. Ashoke De
Department of Aerospace Engineering
Indian Institute of Technology, Kanpur

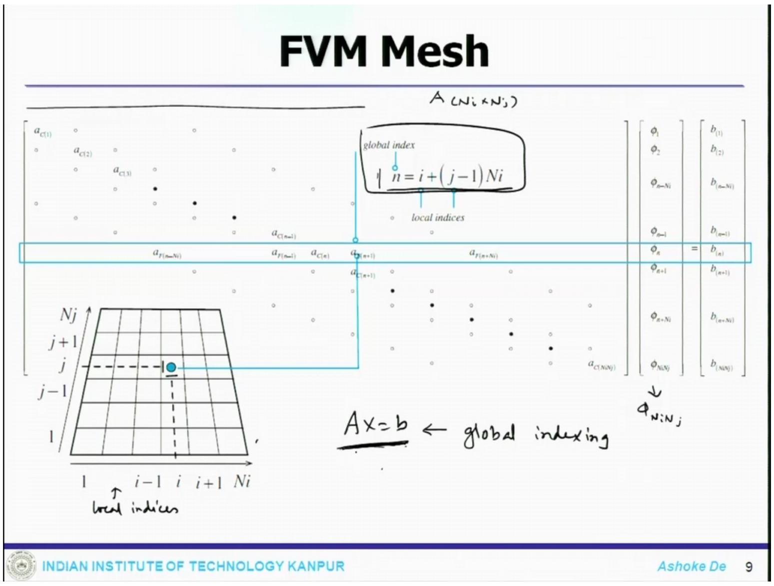
Lecture - 18
Structured Mesh System

(Refer Slide Time: 00:14)



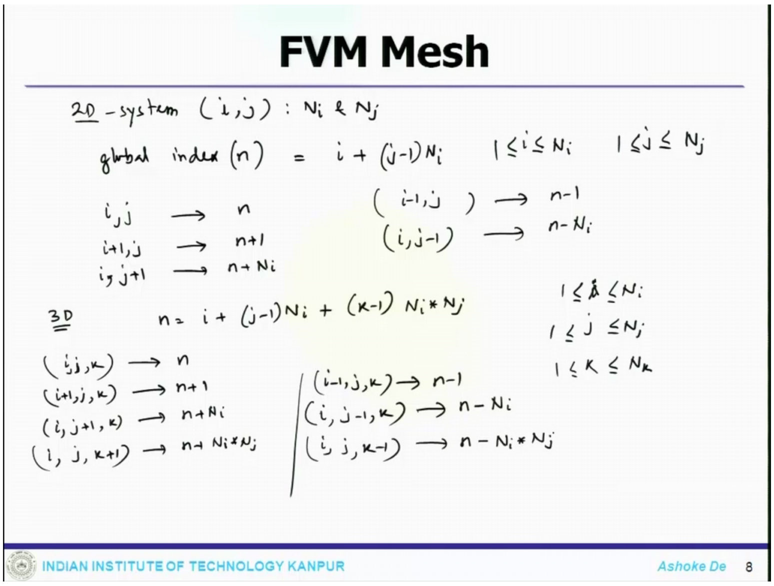
So, welcome to the lecture of this Finite Volume Method. Now given a system one can always find out the, this is a local indices in a i j direction. So, the i j direction can be converted to the global system.

(Refer Slide Time: 00:30)



Now, this is how if you look at it when you have this local indices, this is your local indices. Now local indices is transform back to the global indices and how efficiently it can be done.

(Refer Slide Time: 00:52)



So, for example, if you have a 2 dimensional system, 2D system where you have i and j essentially that corresponds to N i and N j then the global index that can be calculated as n which is i plus j minus 1 into N i, where i goes 1 and j goes N j. So, what does; that means, is that I can have this local indices, but from the local indices in a global matrix, if you look at these global matrix there must be global index. And this is what the vector if you look at it.

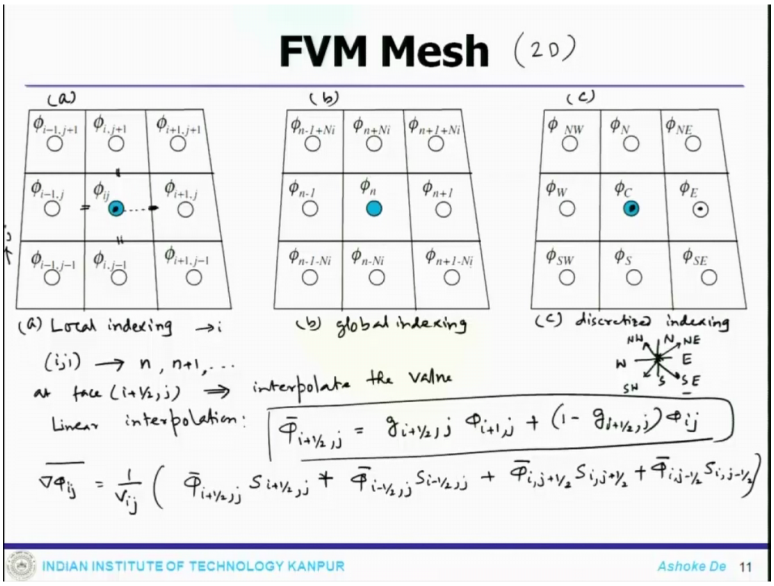
So, 512325 so the local indices must be transform to a global indices for a 2 dimensional system, this is how you transform the local indices to a global indices. Because finally, your linear system has only A which is nothing but your N i into N j and this is going to be 5 N i N j. So now, the mapping from this local indexing to global indexing is very, very essential. So, that my linear solver can handle that; if it is so if i have i j known which can get me the global index n. So, if it is i plus 1 j then it could be the n plus 1 in the global indexing if it is i comma j plus 1, this could be n plus n i. So, if it is i minus 1 j the global indexing would be n minus 1, if it is i j minus 1 the global indexing would be n minus N i.

Now, if you have a 3 dimensional system then you have n equals to i plus j minus 1 into N i plus k minus 1 into N i into N j, where i goes to 1 to N i j goes from 1 to N j, k goes to 1 to N k ok. And they are also similarly one can think any i, j, k they can represent the global index of n. Any i plus 1 the j k the global indexing would be n plus 1 any i j plus 1 k n plus N i, i j k plus 1 this is n plus N i into N j. If you have let us say i minus 1 j k that global indexing is n minus 1. If it is i j minus 1 k the global indexing is n minus N i. If it is i j k minus 1 it will become n minus N i into N j.

So, this is how you actually transform the local indexing for 2D or 3D system to the global indexing. And your linear solver is based on global indexing. So, if you note here is that; that how important is this transformation between local to global indexing and that is where your the definition of the arrays your definition of the data structure, these are become important for programming point of view and this is quite beneficial or handy in case of structure mesh, whether this is not going to be that handy for unstructured mesh.

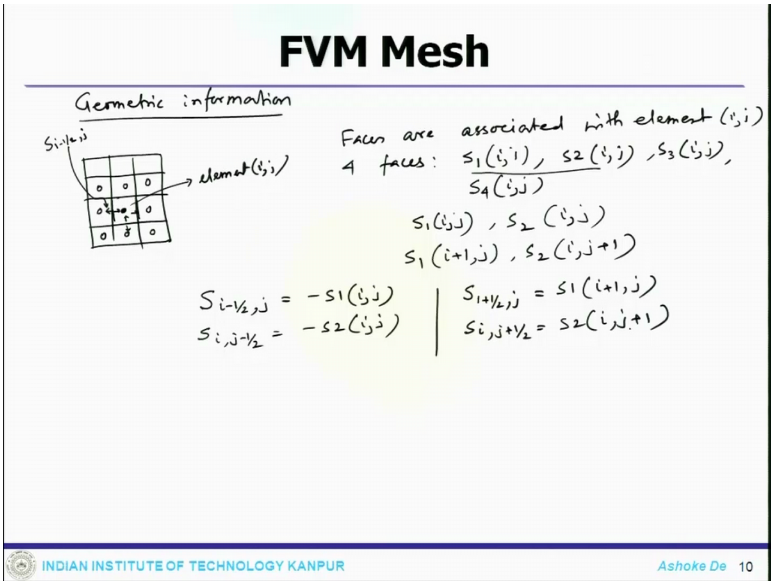
So, that is why this has certain advantage from the coding point of view and a point of view now.

(Refer Slide Time: 05:50)



Now, if you move to the geometric information.

(Refer Slide Time: 00:58)



So, if you store the geometric information then how do you get that or face that in the, geometric information means; if you have a let us say structured mesh like this, and you get this kind of system. So, we have a cell here then you have this, this, this, this, this; these are all surrounding cells. So, let us say this is element i j so the from this particular face there will be 2 component one component is S i minus half j and then there will be component like this component like that.

So, one can use this faces which are associated with element i j. You know the surrounding faces for this particular element would be 4 faces. So, there are 4 faces like S 1 i j, S 2 i j, S 3 i j, S 4 i j ok. Or one can write in other way round that since they are connecting S 1 i j and S 2 i j S 1 i plus 1 j and S 2 i j plus 1 so, because there are some common faces between the elements. So, if you look at those faces then S i minus half j is minus S 1 i j, S i j minus half is S 2 i j. Similarly, if you have S 1 plus half j that is S 1 i plus 1 j and S i j plus half is S 2 j plus 1.

So, that actually gives you an idea the surface vector, how they point to towards each other, now again once you move alone. So, this figure let us say a figure number a, so a actually gives you this is your local indexing ok. If I am interested in this particular element this particular element is sort of surrounded by these elements 1 2 3 4 5 6 7 8 elements and these are my local indexing. This is i j i goes in this direction j goes in this direction. So, if this is the i j ahead of it is a i plus 1 j, downstream of it i minus 1 j, top of it i j plus 1 and underneath of it i j minus 1 this corner it is i minus 1 and j minus 1.

So, it is essentially going this is i plus 1 j plus 1 i minus 1 j plus 1 this is 5. So, all the cell centre values that is stored depends on local indexing of i n j. This is figure b; where if you look at it this is now global indexing; that means, what has been done from this local indexing i j, it has been transformed to n, n plus 1 and so on this kind of global indexing. So, that this indexing would help me to get my linear system properly build up. So, that is why i j has been transformed to cell number n, this side is n plus 1, this is n minus 1, this is n minus N i n plus N i this side if you go n plus 1 minus N i n plus 1 plus n i so and so on.

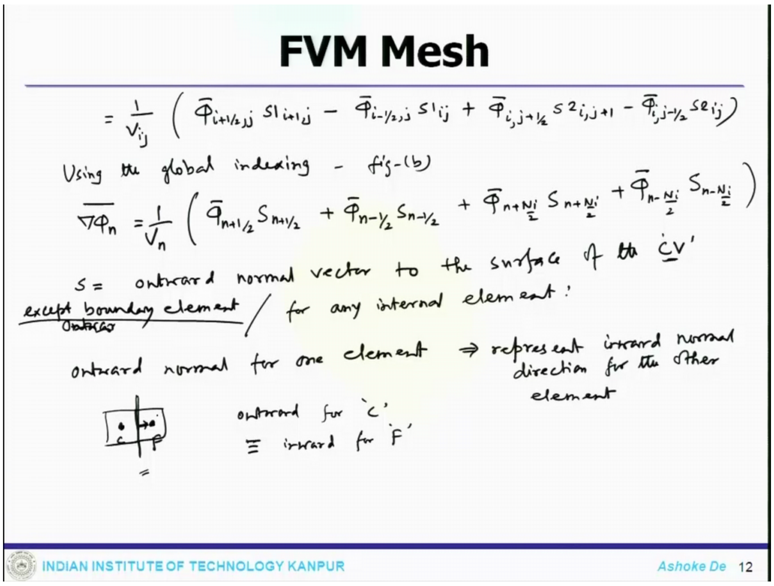
So, this is how global indexing in done. Now this third figure, which is figure c this essentially talks or shows how this particular element around that we are actually writing the governing equation. So, this is essentially your discretize indexing ok. So, you have 3 set of thing that what comes 1 after another, one you have local indexing where you maintain since it is 2 dimensional system. So, we are all talking about 2D system. So, we looking at a 2D mesh so your local indexing involves i j and all these notations and as it is moves along from one cell to another cell. Now from those i j you transform everything to the global indexing of n and n plus 1 like that ok. And then finally, when you look at that particular cell that element c there you write this dicretize indexing, which is again associated with all these elements surrounding that.

And what does that do; the discretize system when you talk about the element c you get east of it. So, the notation is just like an directional notation. You essentially talk like this is my element of interest east, west, north, south this goes north east, this goes north west this goes south west, this go south east this is the very this is one of the interesting or beauty of finite volume notation, when you come down to the local cell especially finite volume discretization of a structured system. So, note that think carefully, it is a structured element that we are talking about there the discretize cells are nicely located and they are indexing is done east, west, south, east and south north, north east, north west.

So, that this goes right or normal direction things. Once you are done that; so what you can do, you can always at face i plus half j. You can find out the or interpolate the value how do you do that. So, you essentially do through the linear interpolation. So, using the linear interpolation you get phi i plus half j equals to g i plus half j phi i plus 1 j plus 1 minus g i plus half j into phi i j. So, this is how you get the linear interpolation done at face and i plus half j face is we talk about this particular face this is i j so this is your face at i plus half j.

So, this is the face we are talking about and any value between these we can actually linearly interpolate and this waiting function one can calculate I mean accordingly. Now the details of this we will see as we move along. So, if you look at the this particular figure a local indexing, one can write this delta phi i j equals to 1 by b i j into phi i plus half j S i plus half j plus phi i minus half j S i minus half j plus phi i j plus half S i j plus half plus phi i j minus half S i j minus half. So, essentially the gradient calculation is done using some sort of a weighted average, taking that interpolated value at here and that surface vector taking the interpolated value here, taking the interpolated value here, taking the interpolated value here.

(Refer Slide Time: 17:08)



So, around those 4 faces by considering that, one can so I can re write that whole thing by V i j which is phi i plus half j S 1 i plus 1 j minus phi i minus half j S 1 i j plus phi i j plus half S 2 i j plus 1 minus phi i j minus half S 2 i j. So, you can actually write that using those local. Now one can also similarly if I using the global indexing so, you refer to figure b; that means, this particular figure where you have actually transformed your i j to N i plus 1 j to n plus 1 i minus 1 j to n minus 1 and so on.

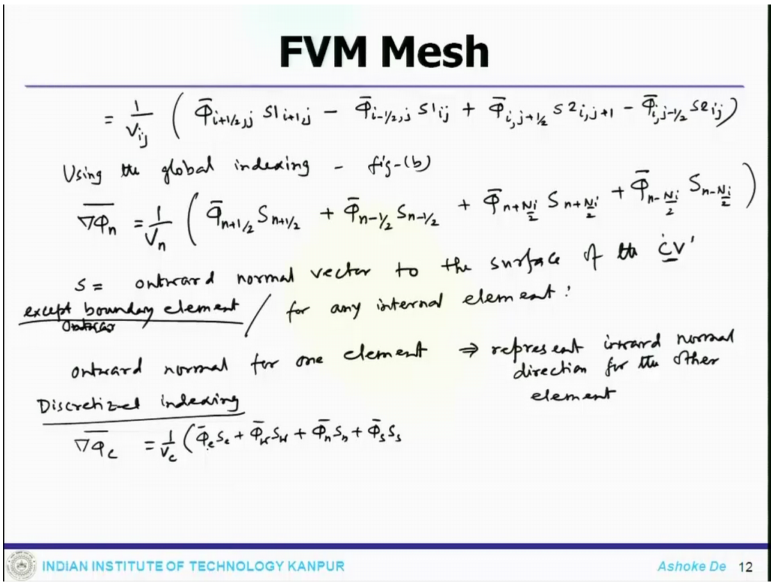
So, if you use those global indexing you can similarly write this is the global indexing I can write the weighted average like this, phi n plus half S n plus half plus phi n minus half S n minus half plus phi n plus N i by 2 S n plus N i by 2 and phi n minus N i by 2 S n minus N i by 2; so, where S is essentially outward normal vector to the surface of the control volume ok. So, you can write all this, so one has to also note that actually the control volumes.

So, it may possible as we mentioned that outward this is except boundary element; that means, if you go back to this figure any internal element not the boundary element any internal element; that means, for any internal element the outward normal for one element will essentially represent the inward normal, inward normal direction for the other element.

So; that means, if I have 2 cell adjacent to each other, and this is the common surface. So, this outward normal could be the inward so for this cell this outward normal could be the inward normal for this. So, if this is C and this is F so outward for c is equivalent to inward for F; so, this is what one has to keep track of it, and at the beginning that is what I have been mentioning that; for any particular cell the normal and the sign is also very important to keep track of it.

Now, another property that if you go down to the discretized indexing. So, far we have local indexing, that is how you get the average quantity or the gradient calculation you use global indexing. So, you get like this, now you can actually use the discretized indexing to get the similar calculation done.

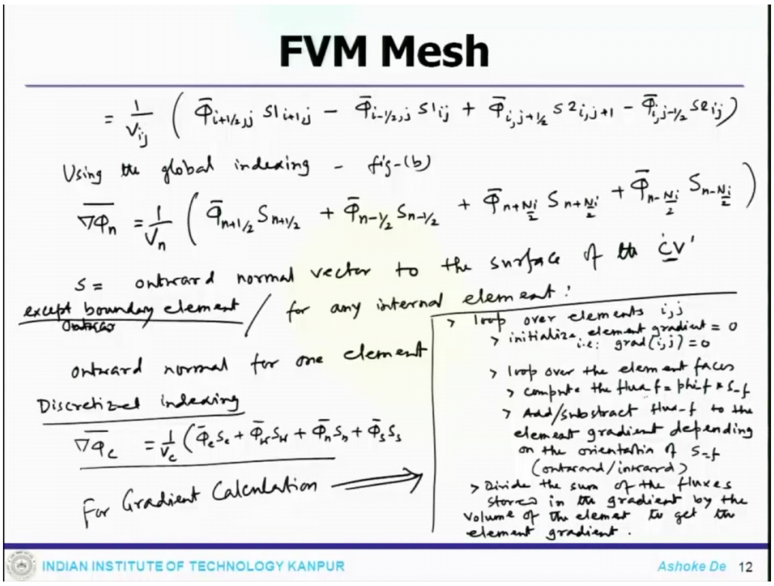
(Refer Slide Time: 22:48)



So, if you look at the discretized indexing so; that means, if you looked at the discretized indexing then you get this gradient calculation of C, which means you go back to this particular picture. This is my element of interest its eastern side there is a element E, north there is a element north, south that is south, south west, west, north west, south east, north east.

So, essentially using these one has to write V C phi e S e plus phi W S W plus phi n S n plus phi S, S S this small e actually represent the face of between C and e this is face e, this is face small n, this is face W, this is S. So, these are all faces which are sort of represented by that number and for this particular cell you write them like this. So, one can have a pseudo algorithm for getting this calculation done or getting the gradient calculation done in this kind of a structured system.

(Refer Slide Time: 24:37)



So, let us say if you write that down so this is for gradient calculation. So, one can find out is a pseudo algorithm, you go loop over elements i j ok. And then you essentially initialize element gradient equals to 0 or that is; that means, gradient of i j is 0 ok. Then you loop over the element faces ok, then what you do; you compute the flux f which is phi underscore f into S underscore f ok. Then you either add or subtract flux f to the element gradient depending on depending on the orientation of S f.

So essentially outward or inward so it takes care of the direction of that so outward normal or inward normal then you divide the sum of the fluxes stored in the gradient by the volume of the element ok, to get the element gradient and you stop. So, essentially what you do? You initially go over all the local indexing that is i j once you go over that. So, you initialize all the gradients to be 0 then you go over all the element faces and then you compute the flux.

That means each faces flux, which is phi into S like this you calculate the flux then you add or subtract depending on the unit normal direction. So, whether S f is positive or negative depending on the orientation; that means, it could be inward normal or outward normal for a particular face and then you have get all the sum of this. Once you get the sum you divide by the element volume and you get the total element gradient. So, this is how you calculate, so we will stop here and carry forward in the next lecture.

Thank you.