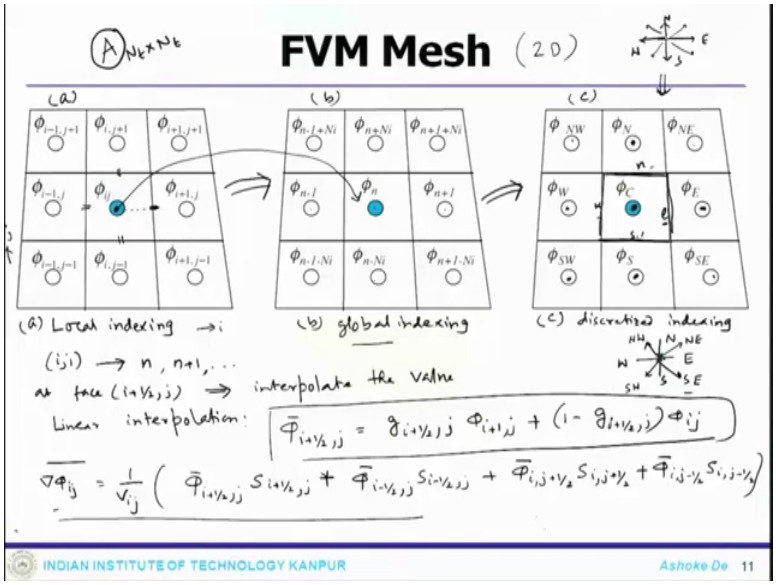
Introduction to Finite Volume Methods-I
Prof. Ashoke De
Department of Aerospace Engineering
Indian Institute of Technology, Kanpur

Lecture – 19
Unstructured Mesh System-I

(Refer Slide Time: 00:15)



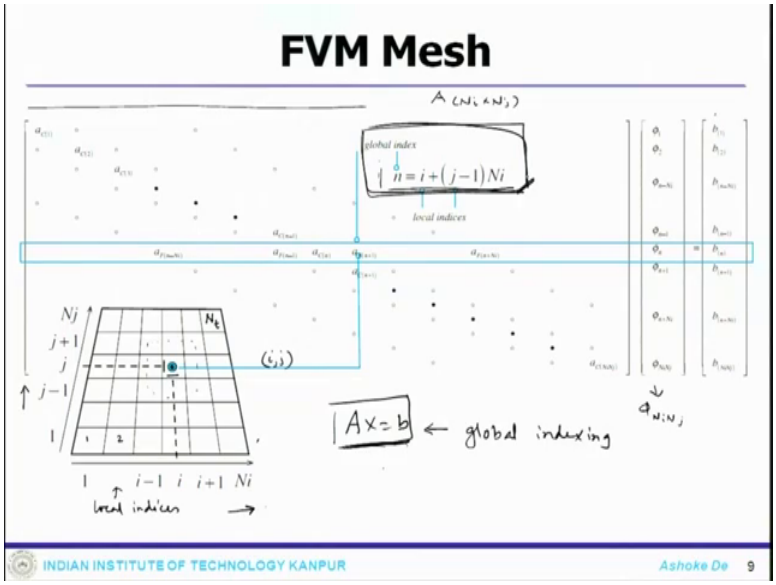
So welcome to the lecture of this Finite Volume Method. So, today we will finish the discussion on the mesh and so, for what we have discussed on the structure mesh, their orientation, how to calculate the surface normal vector how to calculate the gradient and the other terms. So, today we will move to look at the similar things for the unstructured grid system. Now, if you just recall from our previous lecture, this is where we actually stopped we have had a this kind of just to give you an brief overview so, that we can be on the page to continue for the unstructured grid.

So, this is was the structured grid system that we have discussed. So, the figure a which actually was a local indexing, where you have a element at i j location and that element at i j location and then the ahead of it i plus 1 j location, behind that i minus 1 j. So, the local indexing means, you move along the i that and j direction that is essentially in your coordinate system or the referencing system it is a x and y direction.

Once you move along them so, you got some local indexing for the system. Then from local indexing you have we have moved to the global indexing and how it is done? It was

a 2 D system so, the 2 D system it was done in a simpler fashion that i and j count or taken into consideration and then we have done the local to global mapping.

(Refer Slide Time: 02:02)



So, if you just look at this picture which will give you an exact idea, you have a 2 D domain where you moved from i direction and then j direction and you have element at i plus i j location. So, essentially this is your i j location and i j location then you kind of consider this surrounding element to discuss about that this things. From this is where it is a local indexing from local indexing to global indexing is mapped like this so, this is how the global indexing is done.

The reason being your final linear system or the discretize system essentially is going to give you the linear system like Ax equals to b. And the matrix A this only can be handled using the global indexing system and the x vector which will be all the variables for the element from 1 2 to whatever the n number of elements total N number of elements or N t the total N number of elements. So, the variables for the each elements the phi 1 to phi N t these are also in the global indexing system according to the right hand side vector.

So, since the local matrix requires the global indexing system, it is always necessary to transform the so, local indexing to the global indexing. So, that mapping needs to be done and once you do that mapping then you can see the stencil around that particular cell where we are kind of concentrating our discussion that essentially this i j element which is sort of map to this guy in the global indexing system. So, each local index has a

global index and the mapping is done as we have seen right now. So, this mapping actually helps to build the linear system or the global matrix A which would be the total dimension of total number of element by total number of element since it is 2 D.

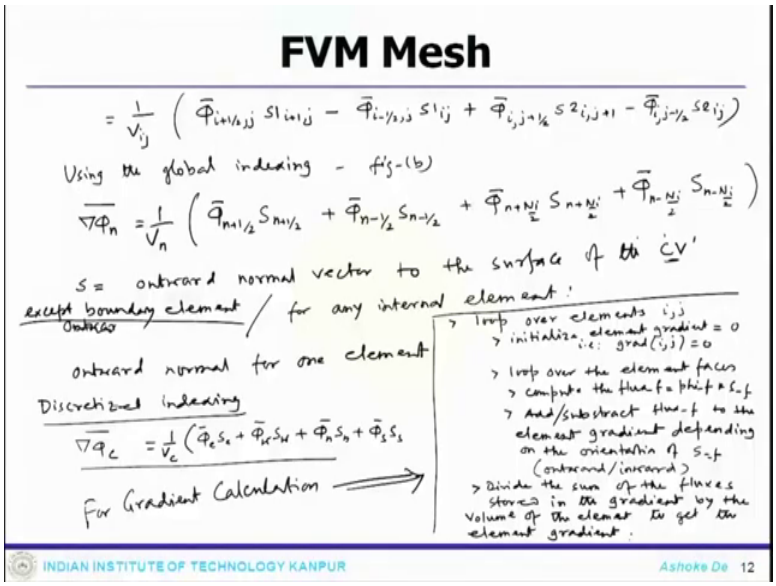
So, each local index has to be mapped to a global index, similarly for this particular example also these element has been mapped to this element, which essentially we call it the nth elements. Then accordingly the other elements which is surrounded by these element and these are the number of elements that requires to be considered or that need to be considered, while doing the discretization for the gradient calculation. So, then the one which is ahead of it which will be n plus 1, the one which is behind this n minus 1 and so on.

So, one is in the upper side, one is the lower side, one is this corner on the. So, this if you recall from a previous lecture this goes in a nice directional so, north south east west and then according to northwest southwest southeast northwest so, this is how you require. Now, third option is that I mean in the sense from local to global indexing so, that helps you to keep track of each element in the system. Now, once you do that you have to get to the discretized system.

So, from there you look at a particular element interior inside the domain or the interior field or any element which belongs to the interior domain which does not belong to the boundary because, the boundary elements needs to be treated separately since there is a specified boundary condition. So, then from there you come down to a discretize indexing, where discretize indexing you can see this direction pattern is kind of represented.

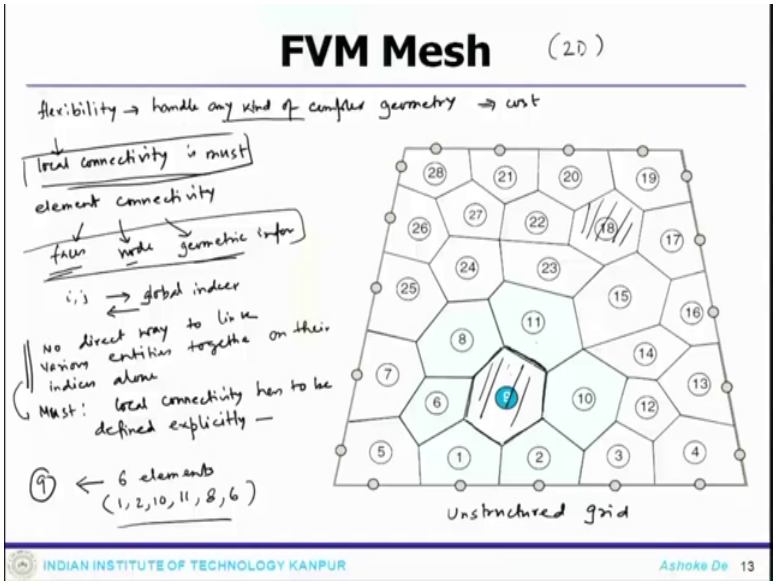
So, something which is ahead of it is east, behind it west upper side north, lower side south southwest northwest and this particular element is connected with 4 faces and the neighboring element like east west north south and those correspondings or connecting faces are marked as a small e small n small west small south. And then we derive the required gradient calculations and finally, we discuss about the pseudo algorithm.

(Refer Slide Time: 06:58)



And if you go over it quickly is essentially you look at the local indexing then you initialize the gradient then finally, you look at each faces and calculate the flux, then as per the sign of the normal vector you or the orientation of the vector you take care of the proper sign of that flux. So, when you add or subtract according to that and finally, you divided by the element volume so, you get the flux. So, that is how is done for the structure system now this is how it is obtained.

(Refer Slide Time: 07:44)



Now, if you move to the unstructured system, this is purely a 2 dimensional again and structured system this is unstructured grid.

Now, why it is unstructured? Because, the elements are not very nicely ordered as we have seen for the structured cases. So, they are sort of randomly oriented, but effectively they must represent the physical domain. Now, what happens to that this gives some sort of a flexibility what kind of flexibility? In the sense that you can handle any kind of complex geometry.

So, that is a added advantage that you have over structured system. Because, the structured system its very specific to sort of a regular Cartesian kind of system, but in the unstructured grid any complex geometry with any sort of curvature cut edges you can easily handled but, it comes with some sort of a cost ok, because, nothing comes at free of cost.

So, when you can handle a complex geometry; obviously, in a unstructured system they are not numbered sequentially so, your local connectivity or the connectivity is must. So, local connectivity information needs to be kept in your data structure. So, that is why it does not come with a free of cost, it comes with some sort of a cost or price because you are handling of data structure become little bit I mean if not cost effective it is essentially some involved handling is required why because since these elements as you see 1 2 3 4 5 6 there all randomly located.

So, you need to take care of that certain all the element connectivity. So, element connectivity in the sense that information of each elements its faces, it is node and other geometric information. So, for a particular element if you consider this particular element so, you need to and also other geometric information which includes all the neighboring cells.

So, one particular element in a unstructured system needs to have all these issues or the information. So, once has to take into account all these information in it is data structure for the programming. So, if you look at or immediately compare between a structure or unstructured, you could see the involvement which required in this kind of systems.

In a structured system it was nicely ordered so, if you just keep track of local indexing like i j from there you can always go back to global indexing and vice versa; that means,

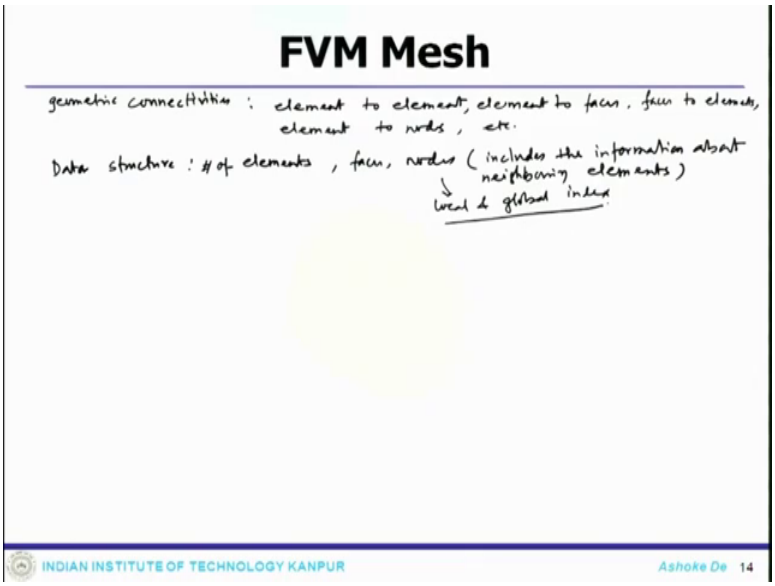
from local to global or global to local transformation is much easier whereas, if you look at in the unstructured grid, that is not like straight forward. So, it requires some sort of an involvement while keeping track of elements connectivity, local connectivity and then accordingly from the local connectivity to the global connectivity.

So, the elements which are surrounded by the other elements you need to keep track of that. So, which essentially comes with some node information, face information and other information so, that means, in a unstructured system there is no direct way to link various entities together on their indices alone. So, this is a very important statement so, that is why in a unstructured system there is no direct way to link that. So, that leads to a must condition is that your local connectivity has to be defined explicitly along with these other information like faces node etcetera.

So, if you just concentrate on a particular element, then we can looking at this complete domain, now you can look at this particular element number 9 and then if you see how things are done that would be give you an I mean exactly or clear idea, how the connectivity and the flux calculation needs to be done for this interior element. So, let us concentrate on a element number 9 and if you look at that element number 9, the 9 is surrounded by another. So, this is the element number 9 is surrounded by 6 elements and those are 1, 2, 10, 11, 8, 6.

So, these 6 elements they are kind of situated or sit alongside with the 9 and there are certain common faces with each of these elements. So, while doing the calculation of the flux one needs to take into account that ok. So, what information one has to do that you have to have some sort of a geometric connectivity's which means your connectivity between element to element.

(Refer Slide Time: 14:45)



So, that means, here 9 which is kind of surrounded by all these 6 elements. So, the information or the geometric information which requires is the connectivity between element to element. Then another important things is that element to faces ok. So, that means if you again come back here this is the element, if 9 is connected with 11 through this common face. So, this face connectivity also needs to be tacked upon, then you need faces to element.

So, which I just said element to faces it is the vice versa of that faces to then element to node; that means, this is the element and then what are the vertices it has so, the node information. So, once you have that then only you know how it is connected with the other element or the associated element along with the which faces because that node information is important and so on.

So, these are the very very important that so, essentially if you look at the programming point of view the data structure. So, data structure needs to be defined for element or rather number of elements faces and nodes and which also includes the information about neighboring neighbouring elements ok. So, essentially these will lead to your local and global indexing so, from here you can have an estimate of your local and global indexing ok. So, we will stop here today and we will take from here in the follow up lectures.

Thank you.