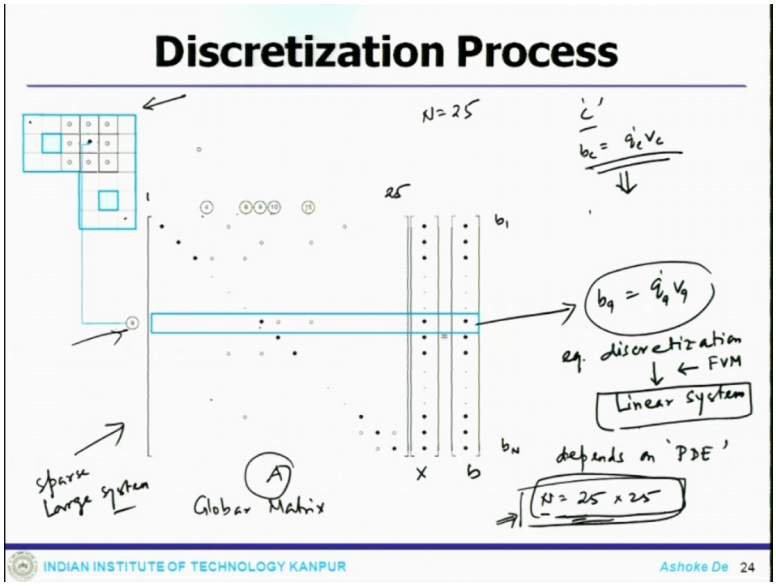
Introduction to Finite Volume Methods-I
Prof. Ashoke De
Department of Aerospace Engineering
Indian Institute of Technology, Kanpur

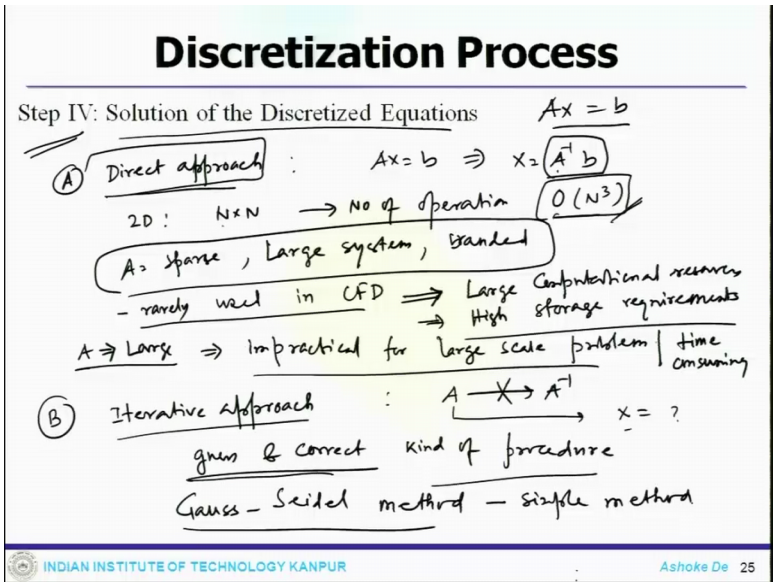
Lecture - 08

(Refer Slide Time: 00:14)



So, welcome to the lecture of this Finite Volume Method. What are the solution procedure that you can adopt? So, that comes under solution of the discretized equation.

(Refer Slide Time: 00:21)



Essentially, this is what I am trying to solve my A x equals to b that is my step 4 or the final step. Now what are the methods. So, one approach could be direct approach. Now direct approach means what do you do? That means, I have Ax equals to b I will directly get x equals to A inverse b. So, what does it mean? That means, I am directly trying to get the inverse of this A, which is not a easy task.

Because you think about it even 2D system, the N by N system, the number of operation that one required to do is order of N cube so, which is large. And sometimes since this system A is not going to be smooth function, or smooth matrix this could be a problem of getting the direct inversion. So, that is why a typically A is sparse large system, ok sometimes it could be banded, ok. So, it depends on the kind of PDEs that we are solving for. Now so, what it does the direct approach is not that we will accept it, ok.

So, what essentially one uses is, I mean not very often the direct approach is used, it is very rarely used in CFD solutions. Because one of the biggest disadvantage of this particular direct approach is that your large computational resources, that is one of the requirement. So, it requires large computational resources, then also high storage. Storage means we talk about in the computer architecture storage in the sense memory. So the ram requirement high storage requirement, because the linear system so, this is where the problem comes. That you require very large computational resources, you require high storage requirement.

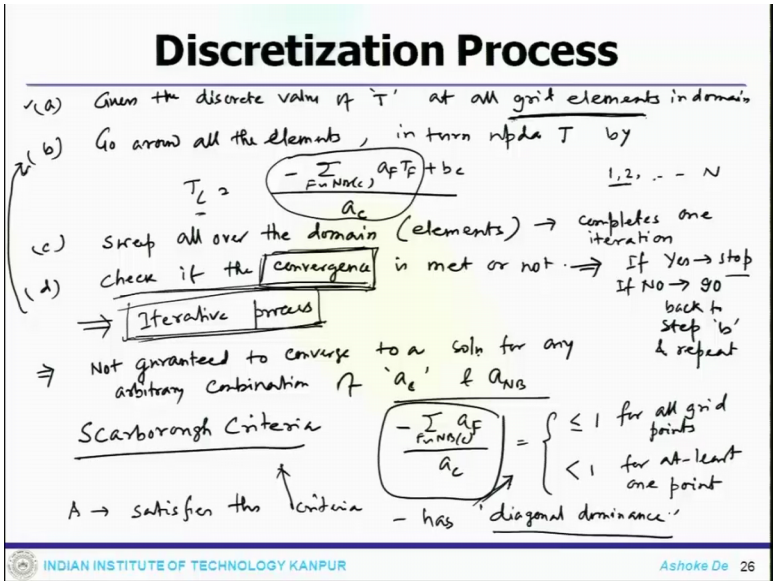
So, that is why very rarely use this CFD in, and top of that the when the size of A, if A is too large, then it becomes just impractical for large scale problem. So, that means, most of our realistic problem are large scale problem. And once you have a large scale problem, that A becomes large and you can get an idea what kind of A even 2D N by N kind of system and the number of operation required order of n q. So, it is quite impractical for the and also it is time consuming. So, it is really time consuming process. So, what is the alternative? Alternative is that people which often used is the iterative process, iterative approach.

So, the iterative approach, I do not get the solution directly, this is not done directly A inverse. Rather, I will do a iterative process get solution for x, ok. So, that is done iteratively. And that works on more like an guess and correct, guess and correct kind of procedure. Because once you try to solve iteratively; that means, you do not have the

exact solution, and you are not trying to get the inverse directly. So, what you try to do? You initially guess some solution try to find out the solution look at the correctness of the solution, you update it. And one of the simple method you can think about is the Gauss Seidel method.

So, what it does? This is one of the simplest of the lot, or it is a very simple method. So, in a Gauss Seidel method or the Gauss Seidel method what it does? The algorithm is very simple.

(Refer Slide Time: 05:12)



You first you do guess the discrete values of T at all grid elements in domain. Here we are talking about element, because we are talking about the cell center value.

So, mind keep that in mind; that we are talking about cell centre value. In this particular approach has nothing to do with the numerical technique that it used. Whether it is a finite difference method or finite volume method, because it can be applied to any numerical method; it is essentially a applied to a linear system. Once you get a linear system you can apply it to any method. So, it does not have any restriction what kind of method you adopt.

So, essentially it first guess the values for this particular example, we are talking about the cell centre values. So, essentially it is assuming in all elements, if you would have a finite difference you get all the nodal points, it will assume in all nodal points in the

domain. So, then you go around all the elements or intern update T by this equation. Like, T c equals to minus F a F T F b c by a c so, you do that. So, essentially, the right hand side of the equation you take everything on the right hand side, then you get this particular system you update that.

So, if you are let us say you are going from 1 2 like that to N, when you come to 3 you may use the updated value of previous ones. When you come to let us say 10th you can use the updated value of the previous ones. So, that it will be more quickly it will reach to the solution. So now, you sweep all over the domain so; that means, you cover all the elements that in the domain all over the domain, that means, all the elements you have you cover all the elements go over it. So, that essentially completes one iteration; ok so, that completes one iteration.

Now you put an convergence criteria. Check if the convergence is met or not, ok. If not, if yes, you stop, if not go back to step b and repeat the process, ok. So, that means, you initially at this step you initially guess the value. So, you start with a guess in all the points in the domain. So, if you look at this particular equation in step b it is essentially taking your all the things at the right hand side and getting directly calculate in T c.

So, once you get that value you sweep over all the domains or all the elements. And once you sweep over all the elements that completes the iteration. And then you check the convergence criteria, ok. If it yes, because this is the criteria that you have provided that how much error you can have and what kind of error you can get? And all these details we will talk. Then first you understand these are the issues that come in picture; that means, once I discretize the governing equation that is where I need the numerical techniques.

So, I just give you an quick glimpse of the finite volume approach or the finite volume way how you can actually discretize to get the linear system. Once you get the linear system you get the solution. So, this is how you can get a solution this is one of the approach.

Now, here we are using another important term of convergence. What do we mean by convergence we will talk about that. So, this is one of the criteria essentially you can think at this particular moment, that it is an criteria to see whether the solution is perfect or not, if that is not, then you go back to step b and repeat the whole process. And that is

why it goes on a iteration and iteration. And this is why this is called iterative process, ok.

Because in one iteration you can hardly expect to get the solution converged. Again I am saying solution convergence. Why, how and what are the criteria? That we will see ok. But one of this procedure the procedure we have described here or discussed here. It is not guaranteed to converged to a solution for any arbitrary combination of a c.

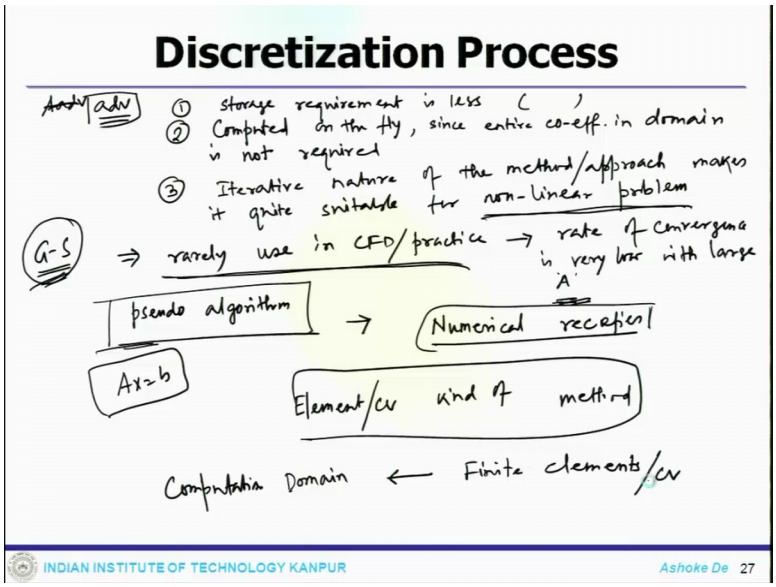
So, important point here is that not guaranteed to converge to a solution for any arbitrary combinations of a c and a NB so, this is a important point. So, any arbitrary combination of a NB and a c it will not satisfied. The criteria that is needs to be satisfied is called the Scarbor criteria, ok. So, one needs to check these criteria and this criteria suggest that minus summation of f where a F is there by a c this ratio, this is the ratio summation of that divided by a c; that means, the first part of the updation process. So, the first component if you see here this must be satisfying some of the criteria. It must be less than 1 for all grid points, ok and it less than 1 for at least one point. So, that means, this is a criteria which is strictly enforce in.

So, that means, when you talk about this any iterative process and which will do. Later on or the latter half of the lectures, that what are the iterative process, what are the iterative methods and different methods, how they are actually evolved, what are the restriction and those restrictions come from certain criterias. So, it is not that once you arrive at Ax equals to b you can get a solution, it is not guaranteed. There are restrictions, there would be restrictions and as a engineer, as a scientist, you are supposed to do those checking or as a programmer you are need to put those criteria in place. So, that you get a solution which will be converged. Otherwise solution may not converge, you might get all junk, ok.

See CFD that is why sometimes people call is a Colorful Fluid Dynamics. Because it can get you all sort of the colorful numbers, but the numbers need to make sense, and as a scientist you are supposed to make sure that those numbers are correct. Now one important things any linear matrix A which satisfy this particular criteria, satisfies these criteria, they has diagonal dominance, ok. If you recall your linear algebra, any linear system if you have a diagonal dominance; that means, the solution is easy to get, and I can get the solution quite easily, ok.

Now, what are the advantage of this particular system? If you put down some of the advantage because since we are talking about there must be some advantage over direct process.

(Refer Slide Time: 14:55)



So, one important advantage is that if you look at it, if you look at this particular process, it just goes on from certain steps. And those steps are nothing but doing certain floating point calculations. I have assumed some value here; I have estimated some value here. I have gone all over the places in the domain. I have checked the criteria, if satisfied I stop if do not I go back. So, these are all essentially floating point calculation. So, the storage requirement is less compared to direct method.

So, when you compared to direct method the storage requirement is least. Second why that is less? Because the whole thing are computed on the fly, because why I can compute on the fly? Again if you come to this particular step, you see what is the requirement for T c; it just required the neighboring elements. Because this is a summation over the neighboring elements, ok and this is what it is. The summation over the neighboring elements so, that makes is quite effective computed on the fly, because since entire coefficient in domain is not required, ok.

So, third is that the iterative nature of the process, iterative nature of the process of the method or rather approach makes it quite suitable for non-linear problem. So, this is one of the biggest advantage of iterative method, because once you look at the direct method,

if the system is non-linear you really have trouble finding the inverse. So, this is one of the biggest advantage that you have in the.

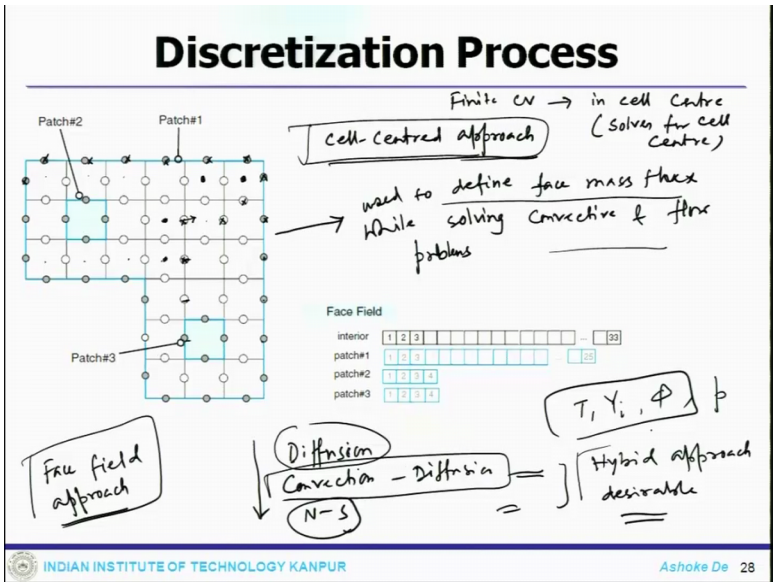
Now, when you talk about particularly the Gauss Seidel method or Gauss Seidel method it is quite really used in CFD, used in CFD or other practice regular. The reason is not that, one of the reason is that rate of convergence is very low with large A. So, when you talk about large A; that means, it is a large scale realistic problem. So, this is one of the drawback of Gauss Seidel method, but as I mentioned earlier this is one of the simplest of the iterative method. So, we just wanted to see how the iterative method works, ok. But this is not the one which is so attractive and used in the real practice.

But it is one of the method someone who is actually getting started with CFD trying to generate or develop his programming code, he can use it very easily. And these days this kind of all this pseudo algorithm, the pseudo algorithm of this linear solvers are essentially available in the numerical recipes, ok. So, numerical recipes if you look at it this is a book written both in C, C plus plus and photron so, you can get the pseudo algorithm.

So, this is not very difficult for someone who is getting started with CFD and getting a linear system. And he can directly solve this linear system using some sort of a direct or iterative solver. But iterative solvers you can get the pseudo algorithm, then only thing that needs to be done is that convert that thing to the particularly in the program, ok.

So now there are other methods like Gauss Jacobi method, successive over relaxation method, this all these different iterative methods are also available, and we would be probably discussing that when we talk about the linear system. Now if you move ahead there are another thing is that; so, the one which we have discussed so far is the essentially the element or control volume kind of method or the discritizations. In the sense, my domain or the physical domain or the computational domain what you call it the computational domain is actually discretized using some finite elements or CV, Control Volume.

(Refer Slide Time: 21:12)



So, another alternative is that called the face field approach. What is that? If you look at the same problem, now when we did the finite element kind of system when we divided in the finite number of elements, we had numbered this element 1, 2, 3 like that to 25. Now this patch 2 patch 3 they are same patch one is the insulated surface. So, face field method is something where you and if you recall in your finite CV, you store the data in cell centre. Or rather your solves for cell center; that means, I have kept my data in each cell center like this. So, this is what I have got the solution using finite element kind of face.

Now, here in the face field approach, you store the data on the face or the centre of the face. So, this marked shown here this actually shows where I am storing the data. So, one approach I had store the data in the centre, that is called cell centered approach, ok. So, that is cell centered approach. Now this face field approach what you have got? You store the data on the centre of the faces; that means, if you look at this individual element or the interior elements they are stored. So, why do you do that? So, this is quite often this kind of a face centered or face field approach, or rather storing the data at the center of the face is used to define face mass flux. While solving convective and flow problem or fluid flow problem, ok.

So, and also if you recall, when we are doing with the cell centre and talking about a interior cell, we had certain fluxes calculated at the or normal fluxes and all this

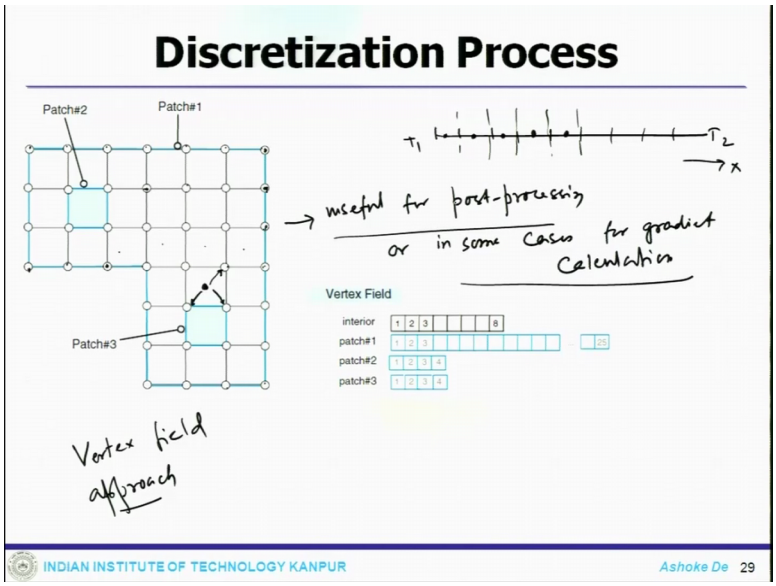
calculated. So, this is quite useful in the face field approach when you store the data at the middle of the cell faces, your solving the convective problem or the fluid flow problem, so, this is quite handy. And another approach could be so this is another alternative, but and when you go along with the system And when you discretize the actually the fluid flow problem. So, initially we will do diffusion problem, do diffusion, then we will add convection diffusion, ok. Then finally, we will go to Navier stokes.

So, once you move along this direction by that time. So, when you solve only the diffusion problem, you will see your cell centered kind of approach could be quiet handy. When you come down to convection diffusion, you may start seeing the effectiveness of face field approach or rather hybrid kind of formulation certain values. You start storing at the cell centre, certain values you start storing at the cell faces so, that is called a hybrid. And when you go down to Navier stokes actually when you go down here this hybrid approach of storing the values would be quite handy, because some scalar values like temperature if you talk about any species mass fraction you talk about any scalar, they are quite handy to store at the cell centre even pressure I would say.

But if you store the velocity at the faces, that would be handy to compute the fluxes and all these thing. That is what I say it will be quite handy to define the mass fluxes. Once you store here the definition of the mass fluxes would be handy to do. That is why when you come down to convection diffusion or to Navier stokes the hybrid approach may be desirable, ok.

But we will also discuss, if you do not do this, what are the problem every approach or mathematics since it is a numerical technique, there could be certain problems. So, we will also discuss about that, what are the pros, what are the cons with self-centered approach, what are the advantage of face field approach, what are the disadvantage of face field approach, what is the biggest gain that one can achieve with the hybrid combination? Ok.

(Refer Slide Time: 26:46)



Now, last one which could be another is the vertex field approach. So now, if you see that vertex field approach, you are now keeping the data at the points. So, that means, this is typically that is what do in the finite difference method. That we store if I think about a straight line and I am getting a solution from here T 1 to T 2 in a one direction. It is just one direction, finite difference just store data in every points, ok. Compared to finite volume, where we store the data in the centers; so, this can be converted into finite volume, this will store the data in the centers. So, this is where the difference is.

Now, vertex field approach is nothing but storing data at the points. So, these are my internal element, and internal element connected with all other element with certain vertex or certain corners. And those corners are now I will store the data. And when it is handy to keep the data there so, these are could be some interior vertices, which could be common to some elements, which could be the exterior vertices which will actually belong like these extra layer or exterior vertices, they belong to the boundary condition.

So, the importance or the this is more useful, this is more useful for post processing the solution data, post processing, ok; that means, when I solve the linear system, I will get a solution when I get the solution. Now post processing of the data would be handy when you use this kind of and so; that means, when I have a cell centered data, I got the solution here. Now I am actually moving the solution to this cell vertices. And if I wrote the vertices, that is more handy for post processing, or in some cases for gradient

calculation, ok. So, this kind of a data storage would be handy. Sometimes it would be handy for gradient calculations or most of the time it is for post processing. So, this is how you get a solution of the linear systems. And we will stop here today.

Thank you very much.