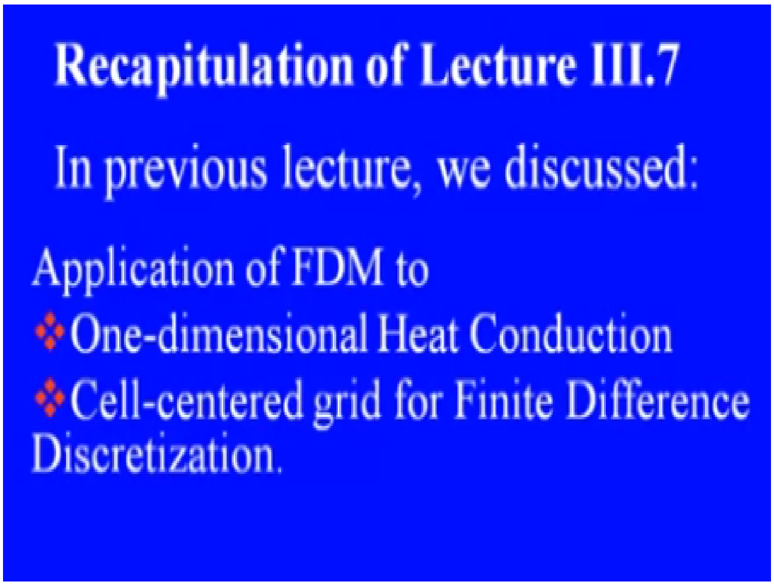
Computational Fluid Dynamics
Dr. Krishna M. Singh
Department of Mechanical and Industrial Engineering
Indian Institute of Technology – Roorkee

Lecture – 17
Applications of FDM to Scalar Transport Problems-2

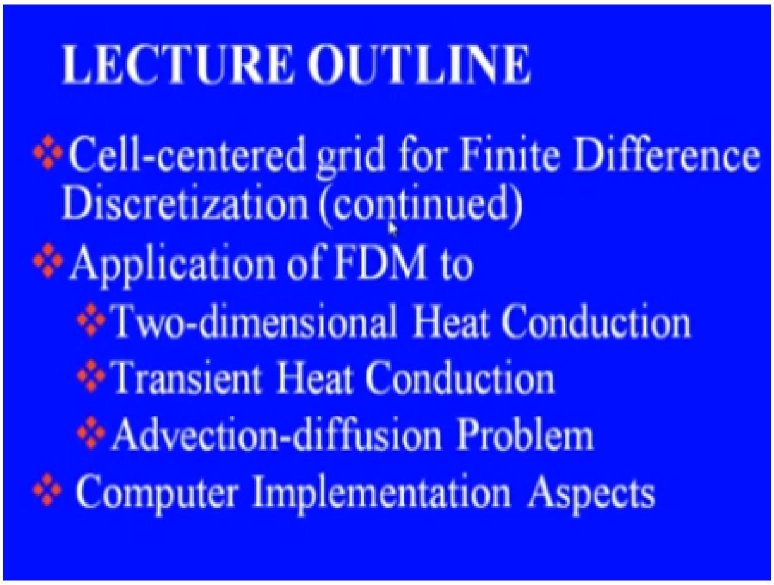
(Refer Slide Time: 00:50)



Welcome once more to our lectures based on application of finite difference method to scalar transport problems. This is our module outline and we had been working on the last topic that is applications of finite difference method to scalar transport problems. In the previous lecture, we discussed application of our difference to one dimensional heat conduction which was based on what we call node or a vertex centered finite difference formulae.

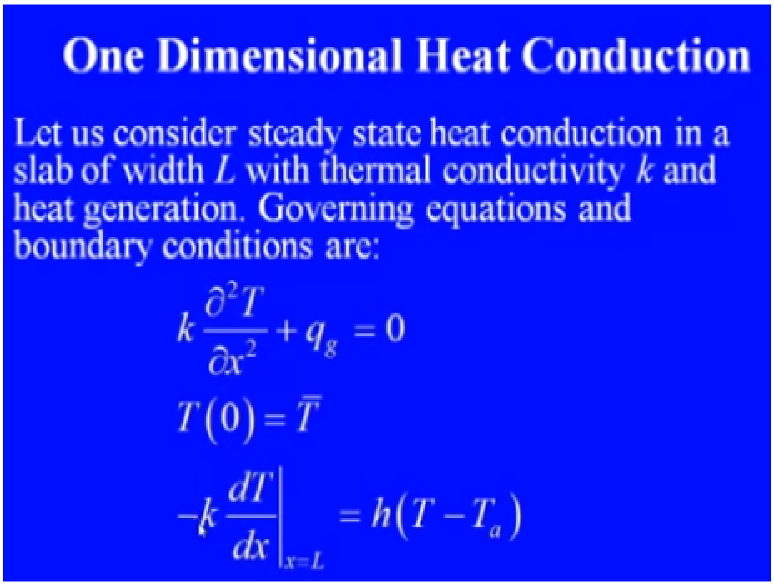
We introduced a cell centered grid for finite differences discretization and we looked at how do we modify the boundary conditions? The boundary conditions have to be taken care of by what we is called ghost cell, we have to incorporate ghost cells from ghost nodes, to incorporate boundary conditions, we did one part that is the incorporation of the Dirichlet boundary conditions and in this lecture this where we are going to continue form.

(Refer Slide Time: 01:48)



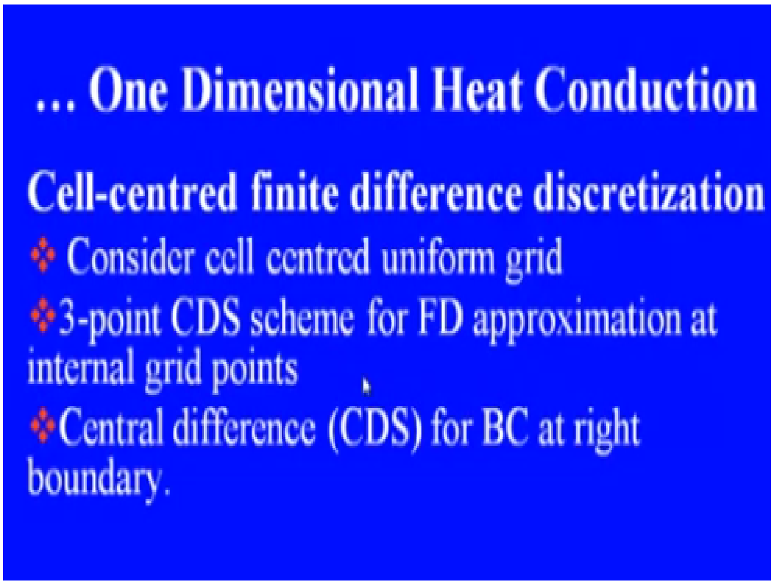
So, this is our second lecture in applications of domain, over all lecture 8 and 5 difference application of FDM to scalar transport problems, part 2, so we will continue with our cell centered grid for finite difference discretization from where we left in the previous lecture and then you would have a look at application of finite difference method to 2 dimensional heat conduction, then transient heat conduction and advection diffusion problem.

(Refer Slide Time: 02:17)



If time permits, we will take up the computer implementation aspect in this lecture, so now let us get back to what we were doing in last class, we wanted to solve it using finite difference this one dimensional heat conduction problem with heat generation, so the governing equation for k del 2T over del x square+qg = 0. At the left edge of the slab, that is at x=0, we had temperature specified and we saw in footway, we can incorporate this temperature specified our Dirichlet boundary condition.

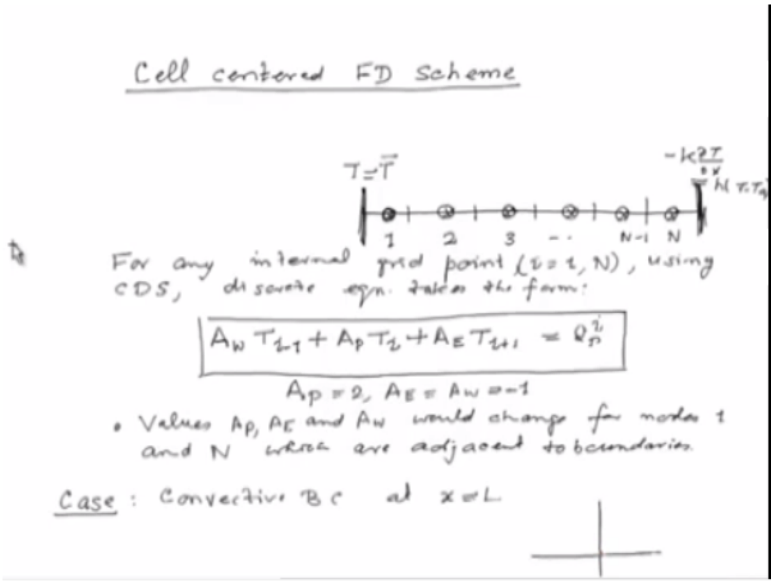
(Refer Slide Time: 03:05)



At the right edge of the slab, that is x= L, we have got a convective boundary condition that is – k dT/dx at x= L= H * T – Ta, so now we need to find out how do we take care of this derivative boundary condition and in what way that is change our discrete equations. So, once again we would continue with the cell centered uniform grid and of course now in this case all our grid points are the nodes they are internal.

So, we would as usual use 3 points CDS scheme for finite difference approximation and we have to modify only this nodes or the equation for the nodes which are close to the boundaries, we have already seen one part and we would again try to use central difference for incorporation of the boundary conditions which involves derivative at the right boundary. So, let us see, how do we incorporate the derivative boundary condition?

(Refer Slide Time: 03:54)

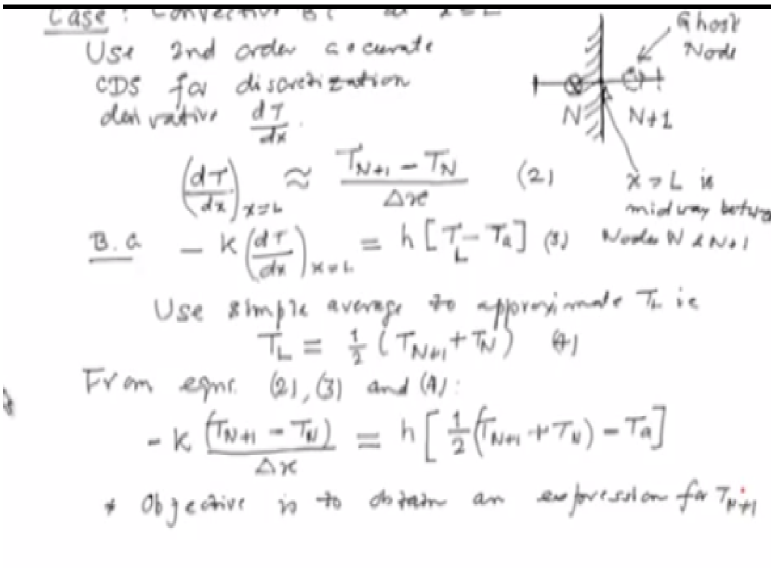


Cell centered finite difference scheme, let us draw our grid, our nodes are located at this centroids of each of these cells node 1, 2, 3 and so on, N-1 and N, this is our right boundary where we have a conductive boundary condition specified and we had at left boundary T = T bar, here we have -k del T/ del x = H * T – Ta, so they mentioned earlier as for; since all these grid points happen to be internal.

So, at each grid point we would replace of governing equation by a discreet counterpart using central difference approximation, so for any internal grid point in fact; please note that now all of our grid points are internal to the problem domain it is i = 1, 2, N using CDS discrete equation takes the form AW Ti -1+AP Ti or TP, for the time being, we are going to prefer using that I index AE T of i+1 and this we said we are going to write QP i.

This is the same as what we had done earlier with the vortex best scheme absolutely no difference, our A, the way rearrange the terms our AP was equal to 2, AE =AW = -1. Now, the values of these coefficients were change for values of AP, AE and AW or change for nodes 1 and N, which are close to boundaries or which are adjacent to boundaries, so we have already seen one change.

(Refer Slide Time: 08:42)



That is we have seen the effect of the Dirichlet boundary condition at node 1, there was Dirichlet boundary condition specified at the left hand and how does it affect these coefficients for node 1 that we saw earlier, so now we will take case 2, that is the incorporation of convective BC at x = L, so once again we would introduce our ghost node, so this is our actual boundary, let us put the internal parts as said it, okay.

This is our internals shell whose middle point is our computational node with the index N, now let us introduce a ghost cell which is outside of it and the centroid of the ghost cell let us put this by dotted line is call or give it an index N+1. Now, with this arrangement a boundary point that happens to be the midpoint between these nodes; actual node N and the external ghost node, so this is our so called ghost node.

So, boundary point is midway, this is a boundary x= L is midway between nodes N and N+1, so that tells us that we can now use of us second order accurate central difference scheme, so let us use second order accurate CDS for discretization of derivative dT / dx, so what do we get? dT/dx at x= L, this we can approximate as T of N+1-TN divided by delta X because the distance between nodes and N+1 would be the same as our normal greater spacing delta x.

Now, let us have a look at our boundary condition, so BC was –k dT/ dx at x = L, this was equal to h times T-Ta, so on the right hand side we have got it on the term, that is our T setting here, so this T is at x= L, so we need to approximate this TL as well. So, let us use simple average, so use simple average to approximate TL that is our TL becomes half of T of N+1+TN.

So, now next we substitute or we use these equations 2, 3 and 4, so from equations 2, 3 and 4 what we will get? On the left hand side, we have -k and approximation for dT/dx would be given by T of N+1-TN divided by delta x, on the right hand side we have got h times 1/2 T of N+1+TN-Ta. Let us try to simplify this expression, our main purpose is to eliminate, so what we are looking for? Our objective is; our objective is to obtain an expression for TN+1.

(Refer Slide Time: 14:03)

... Cell Centered FDM (Derivative BC)

$$T_{N+1} - T_N = -\frac{h\Delta x}{k}\left[\frac{1}{2}\left(\underbrace{T_{N+1}}+T_N\right) - T_a\right]$$

$$\Rightarrow \quad T_{N+1}\left[1 + \frac{h\Delta x}{2k}\right] = T_N\left(1 - \frac{h\Delta x}{2k}\right) + \frac{h\Delta x}{k}T_a$$

$$\Rightarrow \quad \boxed{T_{N+1} = \frac{\left(1 - \frac{h\Delta x}{2k}\right)}{\left(1 + \frac{h\Delta x}{k}\right)}T_N + \frac{(h\Delta x)/k}{\left[1 + \frac{h\Delta x}{2k}\right]}T_a} \quad (5)$$

contains the effect convective BC.

consider discretized obtained using CDS for node N:

$$A_W T_{N-1} + A_P T_N + A_E \underbrace{T_{N+1}} = Q_P^N \quad (6)$$

Eliminate T_{N+1} using eq. (5):

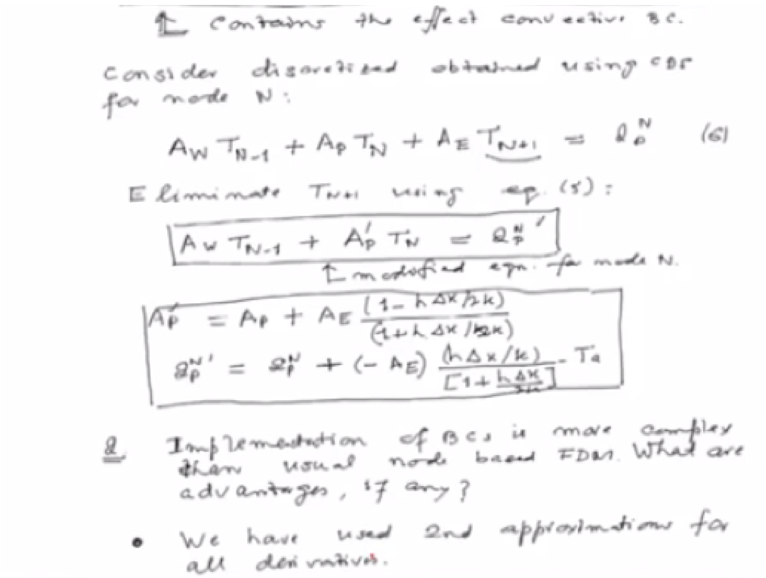
$$A_W T_{N-1} + A_P' T_N = Q_P^{N\prime}$$

This TN +1 is just intermediate variable which we have introduced for our formulation; it has got no other purpose to serve okay. So, we can write our left hand side as T of N+1 – TN, remaining terms let us transfer to the right hand side, so we get -h delta x divided by k and inside we have got by 1/2TN+1+TN- Ta. Let us transfer the remaining term in TN +1 to the right sorry left hand side and bring TN to the right hand side.

So, we get T of N+1 within brackets 1+ h delta x/ 2k, we have just transferred the first term here 1/2 TN +1 multiplied by -x delta x/k to the left hand side, this is equal to TN times within brackets 1- h delta x divided by 2k+h delta x divided by k times Ta, so now this gives us an expression for this intermediate on TN+1 which we have introduced as 1- h delta x divided by 2k, the numerator divided by 1+ h delta x/k multiplied by TN+ h delta x/k divided by 1+h delta x divided by 2k, whole thing multiplied by Ta.

So, now this expression for TN+1, it essentially contains our boundary condition, so this contains the effect of convective boundary condition. So, now let us go to the discrete equation for; consider the discrete equation or other discretized equation obtained using CDS for node N, so this AW T of N -1+AP times T of N+AE times T of N+1 = Q PN, so all that we need to do is now we need to eliminate TN+1 from here using equation 5.

(Refer Slide Time: 19:52)



So, eliminate TN+1 using equation 5 and this will give us AW times T of N-1+, let me write or introduce a new term AP prime TN = QPN Prime, so this becomes our modified equation for node N; node N, where our AP prime would be our original AP+ AE times 1- h delta x divided by 2k divided by 1+ h delta x/k; sorry this should be 2ks everywhere and similarly our QP and Prime is the original value of QPN+ the additional term which comes from the effect of convective boundary condition.

So, it would be-AE times okay, now you transferring the contribution to the right hand side h delta x/k divided by 1+ h delta x divided by 2k times Ta, so what we can say is with cell centered finite differences, the light became bit difficult the implementation boundary condition that required some effort but there is a foot, translate into any gain in computational efficiency or numerical accuracy, so this is what the questions we need to ask ourselves.

The question is implementation of BCs; of BCs is more complex than usual node based FDM, so what are advantages if any? We have spent some good amount of effort on the implementation of these boundary conditions, so it will complicate our coding effort as well, we will require few more computations so what is the gain? One thing which you can guess, we have used, so note here, we have used second order approximations for all derivatives okay.

(Refer Slide Time: 24:22)

Perceived advantage:
* Hopefully, we should get more accurate numerical solution.

<u>System matrix with cell-centered scheme</u>

$$[A] = \begin{bmatrix} A_P^M & A_E & & & \\ A_W & A_P & A_E & & 0 \\ & \ddots & \ddots & \ddots & \\ & 0 & & \ddots & \ddots \\ & & & A_W & A_P^{N''} \end{bmatrix} \qquad \boxed{A_E = A_W = -1}$$

* Matrix A obtained from cell-centered FD discretization is "symmetric"

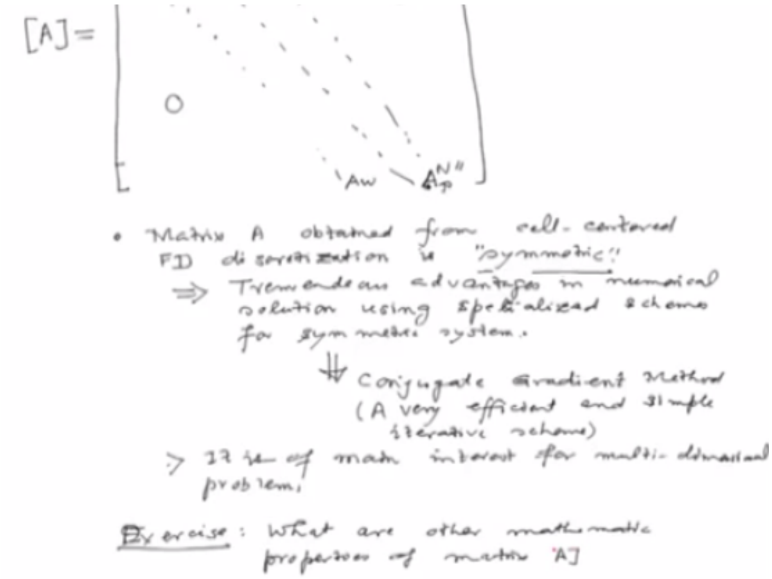
We use central difference scheme for the discretization of d2 T/ del x square term in our governing equation and we also used a central difference approximation for evaluation of dT/ dx term in the boundary condition, so what we said? Is; first thing is perceived advantages hopefully, we should get more accurate numerical solution either any other benefit, for that let us have closer look at the complete system when we assemble.

So, let us write our just system matrix, so system matrix with cell centered scheme, if you remember the case one, what we had they were change only in a diagonal term that is AP, AW dropped out, so the first row we are left with 2 terms that is AP one, let us call it, use a symbol prime or rather to make it clear let us call it APM that is modified coefficient M at node 1 and then usual value of AE, now this AE would be the same for remaining terms.

In the next row, we get AW, then usual value of AP and AE, so this is our system matrix A and so on, we can keep writing it, we get only these 3 diagonal terms, the rest of the terms would be zero. Now let us come to the last row which corresponds to node N, so what we found from the previous equation which we had derived, they were changed only in AP, A will not adjust, this is a last diagonal, so we will get AP and double prime.

So, this is a modified term AW remains same and note that the way we have written our AE and AW they are both equal and their value was -1, so what you can see from this matrix, now this matrix is; see matrix A obtained from cell centered finite difference discretization is symmetric, so this is stark contrast with our previous formulation, where our nodes were aligned with the boundary.

(Refer Slide Time: 28:42)

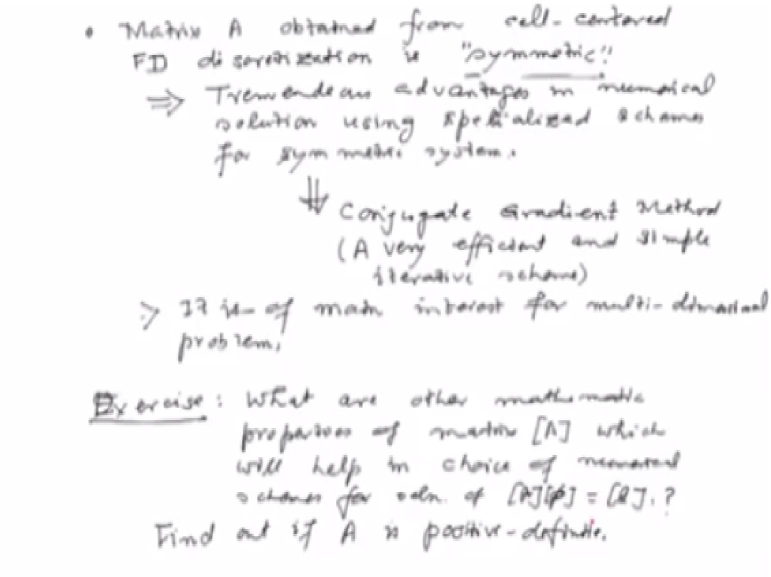


The first node was in the left boundary, the last node that is N+1 and that node was on the right boundary, there we saw that system matrix is not symmetric but here we have got a symmetric system matrix which has got scores of advantages in numerical solution okay. So we have got tremendous advantages in numerical solution because we have got specialized schemes for symmetric systems, using specialized schemes for symmetric systems.

For instance, if you want to use an iterative scheme you have got a very beautiful reiterating scheme which is applicable that is our conjugate gradient method which is a very very simple and it very powerful iterative scheme, a very efficient and simple to program iterative scheme and of course this advantage or like this observation is pertinent for the case of multidimensional problems, for one D problem where we have got only a tri diagonal structure.

We are happy with TDMA; TDMA matrix is symmetric or not symmetric, so this advantage is basically it is of main interest for multidimensional problems though we derived or we showed this matrix to be symmetric only starting with a one D problem but exactly same structure we would obtain if we had a 2D or 3D problem, if you want you can verify it yourself and the procedure for implementation of boundary conditions would remain exactly the same which we have discussed for all our simple one D case.

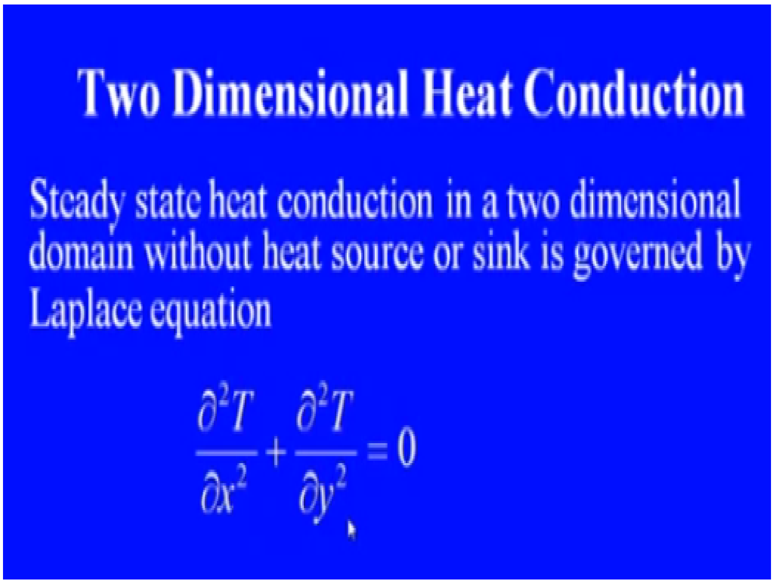
(Refer Slide Time: 32:10)



I would like to leave certain things as an exercise for you, I just mentioned this matrix is symmetric. Can you check with a linear algebra book, what are other mathematical properties of matrix A which will help in choice of numerical schemes for solution of our A phi = Q this system which we had, we want to solve it numerically, so are there any other properties which of; which would be very helpful.

In particular, find out, if A is positive definite okay, I will just close this cell centered approximations with a remark that this scheme we use exclusively in the solution of Navier stokes equations, so we will have what we call a pressure Poisson equation, pressure elliptic equation and for the solution of pressure Poisson equation, we will choose pressure nodes to be at the centres of our cells.

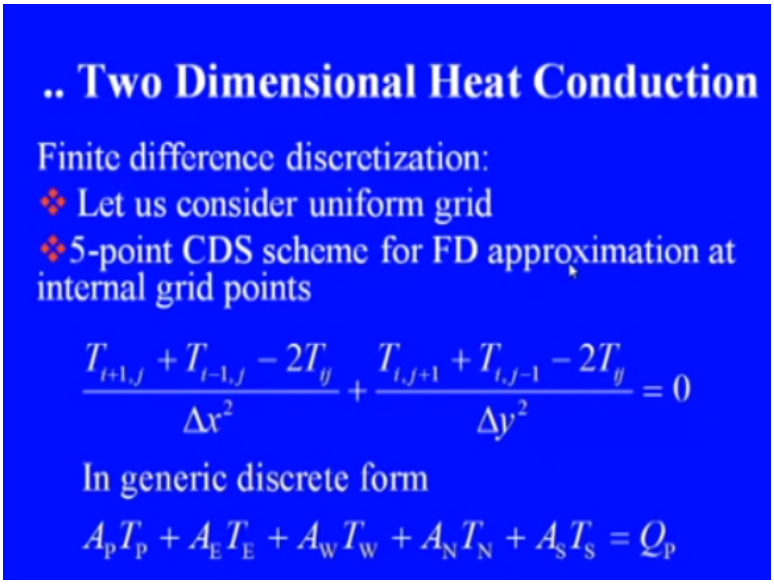
(Refer Slide Time: 33:45)



And the resulting system matrix would be definitely symmetric and that will help us in the solution for pressure Poisson equation. Now, let us have a look at a two dimensional heat conduction problem and see what sort of discretization we can do for this. So let us consider for the sake of simplicity a steady state heat conduction in a two dimensional domain without heat source or sink.

If the heat source or sink is present that can be taken care of very easily, the way we have seen with one dimensional problem, so this particular problem where there are no sources or sinks is governed by our Laplace equation that is delta 2T/ del x square+del 2T divided by del Y square = 0, so you have got second order derivatives and we have already learned multidimensional formulae for second order derivatives.

(Refer Slide Time: 34:34)

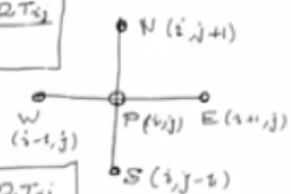


So, what do we do? For finite difference discretization, we will consider a uniform grid of the size delta x in x direction and delta y in y direction and let us adopt a 5 point CDS molecule for finite difference approximation at the internal grid point, so if you do that what happens? Now let us draw our computational molecule to clarify the things a bit now.

(Refer Slide Time: 35:11)

Two dimensional heat conduction

$$\boxed{\left(\frac{\partial^2 T}{\partial x^2}\right)_{ij} \overset{CDS}{\approx} \frac{T_{i+1,j} + T_{i-1,j} - 2T_{ij}}{\Delta x^2}}$$



Similarly,

$$\boxed{\left(\frac{\partial^2 T}{\partial y^2}\right)_{ij} \overset{CDS}{\approx} \frac{T_{i,j+1} + T_{i,j-1} - 2T_{ij}}{\Delta y^2}}$$

Discretized equation at internal node P.

$$T_{i+1,j} + T_{i-1,j} - 2T_{ij} + \frac{\Delta x^2}{\Delta y^2}\left(T_{i,j+1} + T_{i,j-1} - 2T_{ij}\right).$$

Two dimensional heat conduction and what we are doing the way, we are proceeding now that can be easily extended to 3 dimensional problems, so 5 point CDS molecule we are just dealing with the stencil, so these nodes could be at the cell centre, so that these could be at the edges, we do not need really need to worry about it, so our node P ij system neighbour Ei+1, j western neighbour Wi-1,j its northern neighbour in y direction, it will be ij+1.

And then this is for southern neighbour would be given as ij-1, so at point P, that is at point given by the indices i and j, our del 2 T/ del x square, remember what we said if you want to write I want to find out the approximate shown formulae for multidimensional derivative, look at the variable with respect to which we want to differentiate. So, it here it is x; x means only I indices will change in our formulae in which we had derived for one dimensional problems.

Suppose, here we were using CDS, so we will use central difference scheme; P is our central node valued at W and E and P those are the ones which would be involved in approximation of the derivative del 2T over del x square. It is simply given as T i+1j that is the value at eastern neighbour+Ti-1, j value at the western neighbour -2 Tij divided by delta x square. So, this is the approximation for the first second order derivative in our governing equation.

Similarly, the second derivative which we had del 2T/ del y square at the grid point I, j but again use your CDS approximation, so now the values at grid point P N and S, those will be involved because now we have to differentiate with respect to y, so i indices will be kept same in our numerator terms only j indices will change, so T i, j+1+Ti, j-1-2T ij divided by delta y square, then what would be the discrete equation at the node P?

(Refer Slide Time: 40:18)

$$\boxed{\left(\frac{\partial^2 T}{\partial y^2}\right)_{i,j} \overset{CDS}{\approx} \frac{T_{i,j+1} + T_{i,j-1} - 2T_{i,j}}{\Delta y^2}}$$

S (i, j-1)

Discretized equation at internal node P.

$$T_{i+1,j} + T_{i-1,j} - 2T_{ij} + \frac{\Delta x^2}{\Delta y^2}\left(T_{i,j+1} + T_{i,j-1} - 2T_{ij}\right) = 0$$

Let $\beta = \frac{\Delta x}{\Delta y}$

$$\Rightarrow \boxed{-T_{i+1,j} + T_{ij}\left(2+2\beta^2\right) - T_{i-1,j} - \beta^2 T_{i,j+1} - \beta^2 T_{i,j-1} = 0}$$

$$\Rightarrow \boxed{A_W T_W + A_P T_P + A_E T_E + A_N T_N + A_S T_S = Q_P}$$

$A_W = A_E = -1, \quad A_P = 2(1+\beta^2)$

$A_N = A_S = -\beta^2$

- Nodes at boundaries handled e.

So, discretized equation, substitute the approximation of these derivatives in governing equations, so we get discretized equation at internal node P and this would be given by Ti+1, j+Ti-1, j – 2Tij, multiplied by delta x square to simplify the things of it, so we get a ratio of delta x square/ delta y square * T of ij +1+ T of ij-1-2T ij = 0, we can introduce a shorthand notation, so let us the symbol alpha or rather let us use beta.

Beta = delta x divided by delta y, so we can write the preceding quiz in bit more compact form as T of i-1, j+T ij within brackets 2+2 beta square-T of i-1, j-beta square times T of ij+1- beta square T of ij-1=0, so this can be easily recast in our generic form that is AW TW+ AP TP+AE TE+AN TN+AS TS = QP, we can easily compare the coefficients and write these values here that AW= AE = -1, our AP= 2 times 1+ beta square and AN = AS=-beta square.

(Refer Slide Time: 43:29)

$$- 2T_{ij}) = 0$$

Let, $\beta = \frac{\Delta x}{\Delta y}$

$$\Rightarrow \boxed{-T_{i+1,j} + T_{ij}(2+2\beta^2) - T_{i-1,j} - \beta^2 T_{i,j+1} - \beta^2 T_{i,j-1} = 0}$$

$$\Rightarrow \boxed{A_W T_W + A_P T_P + A_E T_E + A_N T_N + A_S T_S = Q_P}$$

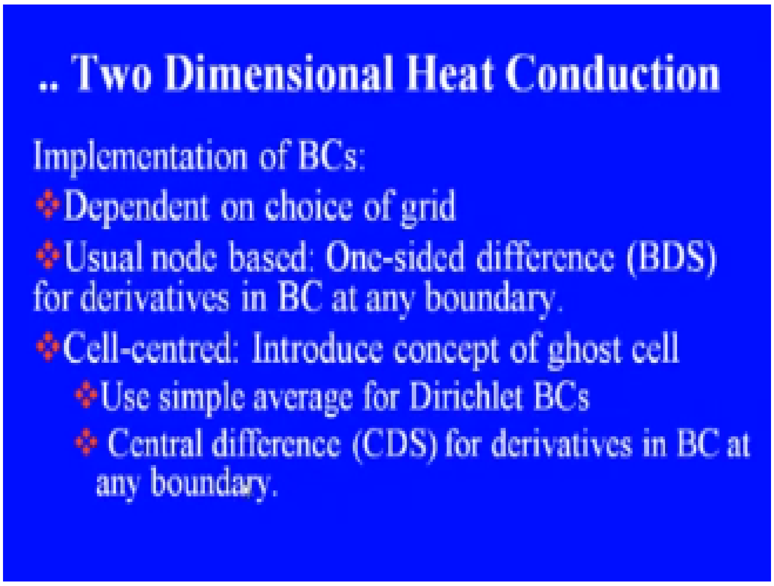
$A_W = A_E = -1$, $A_P = 2(1+\beta^2)$

$A_N = A_S = -\beta^2$

- Nodes at boundaries handled exactly in the same as we did for 1-D problem to incorporate boundary conditions.

Handling the boundary conditions is exactly the same as we had done previously in the case of one dimensional problems, so nodes at boundaries handled exactly in the same way; the same way as we did for one D problem to incorporate boundary conditions and once you have incorporated the boundary conditions, we can easily obtain discrete system of equations by assembling the things at all the nodes.

(Refer Slide Time: 44:30)

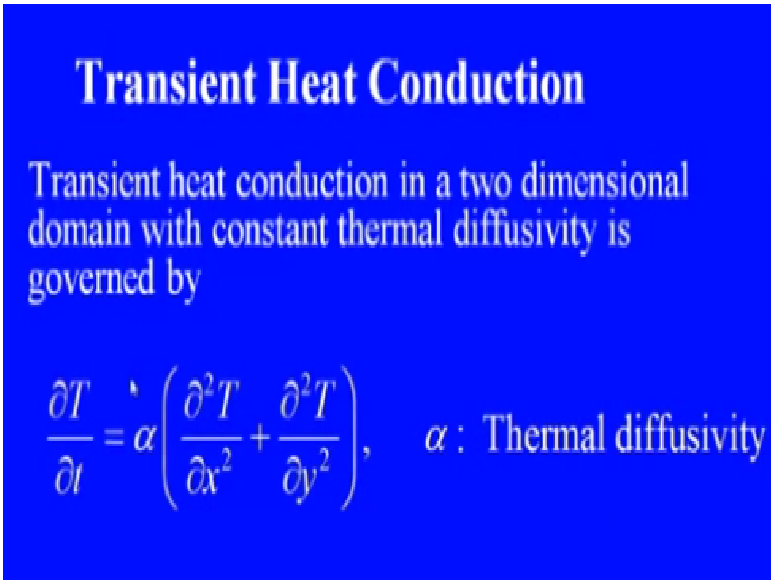


And you can observe for yourself that we will get a penta diagonal system okay, this is what we had our generic discrete form for two dimensional heat conduction problem, so implementation a BC that will depend on our choice of grid that is to say whether we have taken vertex centre computational node or a cell centered computational node. So, if you have got usual node base that is vertex best schemes.

We have to use one sided difference formula BDS or FDS as appropriate for derivative boundary conditions at any boundary. Dirichlet slip boundary conditions are taken care of without any problem, for cell centered scheme, we will proceed exactly the way we did in one D, that is we will introduce the concept of ghost cell and we will use simple average for Dirichlet basis as how we will incorporate and eliminate the values corresponding to the ghost cell.

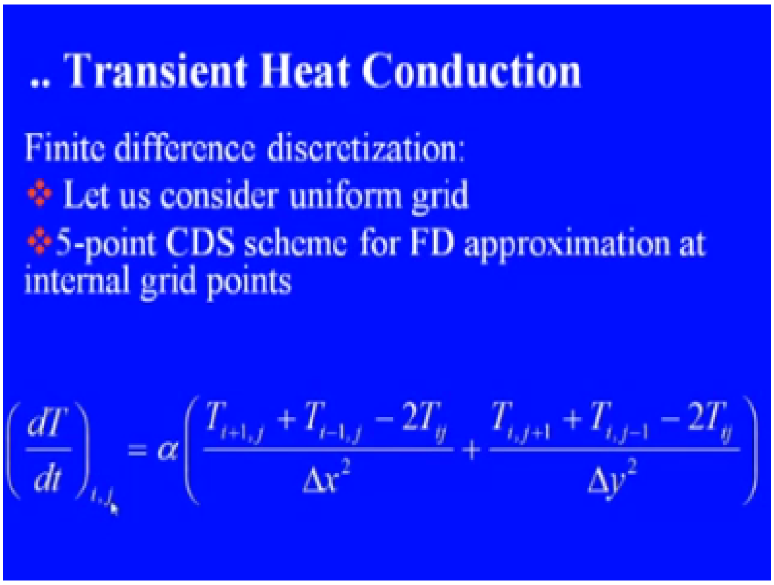
And thereby include the specified boundary condition in the discrete equation for the node close to Dirichlet BC. Similarly, we can use central difference scheme which involves value of the variable at the ghost cell, use your boundary condition to obtain that expression for that variable at ghost cell, substitute that in this discrete equation for the node adjacent to this derivative boundary and we are done with introduction of boundary conditions.

(Refer Slide Time: 45:49)



Next suppose, we had the instead of steady or problem where transient one, so say for example a transient heat conduction problem in a two dimensional domain with constant diffusivity in the the case of variable diffusivity can be taken care of algebra it would become bit more involved but as far as finite difference approximation is concern and these steps would be fairly similar.

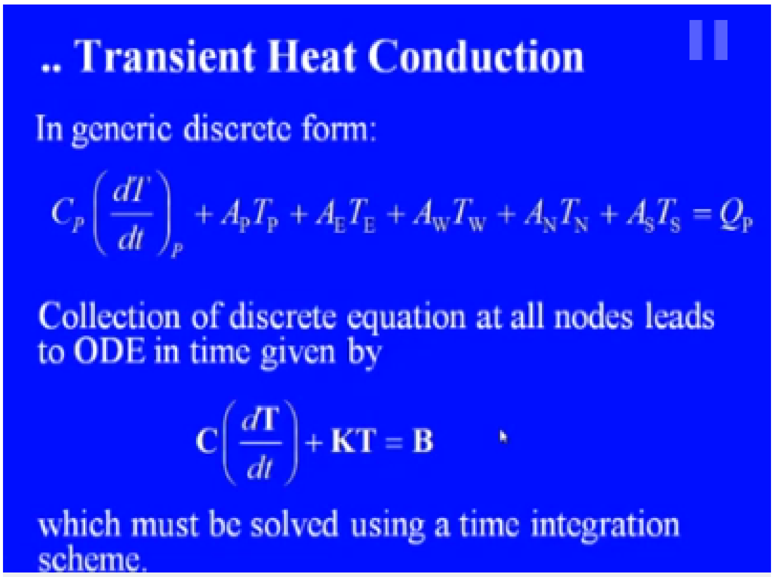
(Refer Slide Time: 45:42)



So, now let us take the case of constant diffusivity, so del capital T/ del T, very small Ts of time variable is equal to alpha times del 2T/ del x square+del 2T/ del y square, if we had a 3 dimensional problem, we will have one more term del 2T/ del z square and here where alpha denotes thermal diffusivity, so for finite difference discretization let us consider a uniform grid of grid size delta x and delta y, 5 point CDS scheme for finite difference approximation at internal grid points.

So, this is what we have already seen earlier for 2D case; 2D, 3D case we derived this as a central difference approximation. What happens in the left hand side? Now, del as a partial derivative of capital T with respect to time now that becomes an ordinary derivative, so that is why I have used symbol d; dT/ d small t i, j at grid point ij = alpha times this bracket terms Ti+1j+Ti -1j -2 T ij divided by delta x square+Ti, j+1+ T I, j -1 -2 T ij is divided by delta y square.

(Refer Slide Time: 47:44)



So, in generic discrete form, we introduce our symbols AP AN and so on and you can write by introducing a new symbol CP for the time being of course was the way we have written it CP =1, so CP dT/ D capital T/ dT at point P with+AP TP+AE TE+AW TW+AN TN+AS TS= QP, this QP might be present in case if we had some stroke terms coming from somewhere.

Now, if you collect such discrete equation at all nodes, so in this case we will get an ordinary differential equation in time, so all these Cs coefficients they will be combined in this matrix C, so this capital C matrix is basically a diagonal matrix, so capital C times d capital T/ dT+capital K * T = B; now these bold symbols, bold capital T indicates or it is a vector of the temperature values at all the grid points.

And capital B denotes what we call the load vector, so in finite difference finite volume of finite element terminology specifically in the case of finite elements this matrix C is also referred to what we call a capacitance matrix, matrix k is called a stiffness matrix and B is called the load vector, so same term is also borrowed and used very often in finite difference and finite volume literature for transient heat conduction problems.

So, this is a simple system of ordinary differential equations one; it contains only first derivative and this has to be solved using a time integration scheme which we will learn in a later module, so until then let us wait and learn few more techniques of finite difference discretization for convective transport problem which we will do in the next lecture and we will discuss some computer implementation aspect also in the next lecture.

And then we will take one by one; the first we will consider in the next module the solution of algebraic equations and then in the module after that we will take; take up the case of solution of these ODE or ordinary differential equations in time