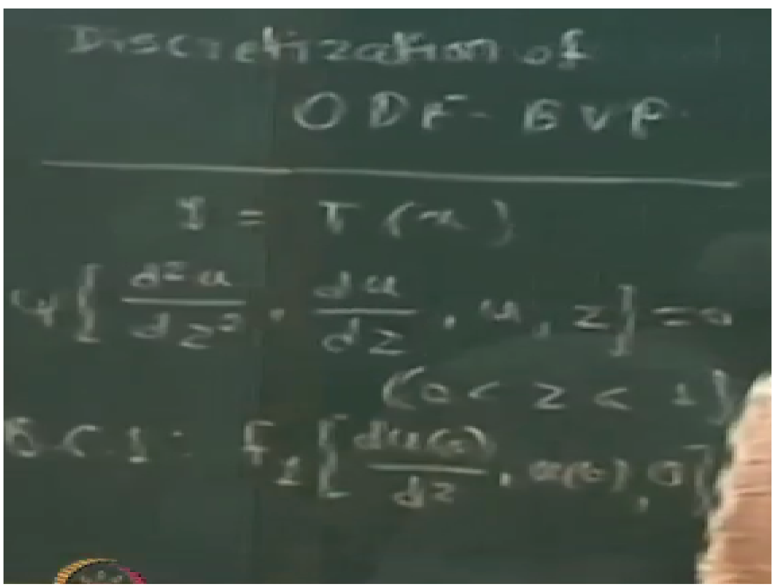
Advanced Numerical Analysis
Prof. Sachin Patwardhan
Department of Chemical Engineering
Indian Institute of Technology – Bombay

Lecture - 13
Solving ODE-BVPs and PDEs using Finite Difference Method

So in our last lecture, we looked at the way of using Taylor series approximations to discretize ODE boundary value problem. So you are able to convert a differential equation and a 2 boundary conditions into a set of algebraic equations. So the problem got transformed from the original infinite dimensional space to a problem on a finite dimensional spaces which was set of linear or non-linear algebraic equations which have to be solved.

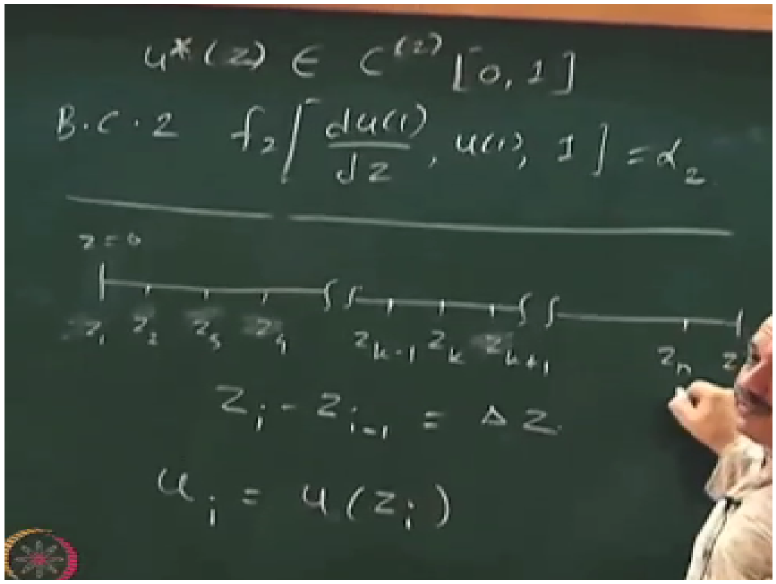
So today let us look at some examples and let us see whether we can extend this concept to solving partial differential equations. So I wrote a generic boundary value problem.

(Refer Slide Time: 01:14)



Discretization of-- So we have this original problem was or I would say y = T of x. This original problem was, so you have this differential equation which is which holds over entire domain 0 to 1. This is the differential equation and then you have 2 boundary conditions one is at—so you have boundary condition 1 which is f1.

(Refer Slide Time: 02:00)

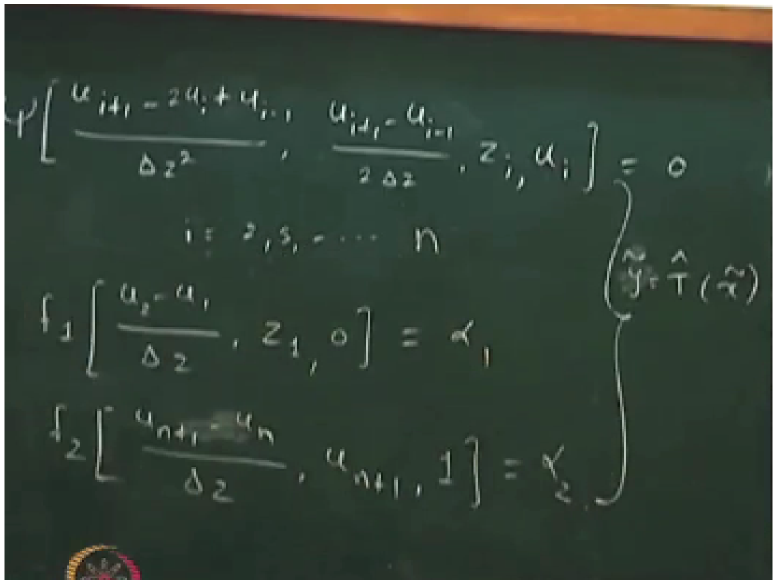


And you have second boundary condition which is f2. You have 2 boundary conditions. The way we proceed next will be discretize the domain. We discretize the domain, we mark this grid points such that using equidistant grid points we discretize the domain, they mark the grid points and then we had this notation that the dependent variable to u, ui = u of, u was the value that this dependent variable takes at z = zi that we denoted as ui.

Now this u here is the approximate solution. The real solution of this would be some u* z which should be a continuous function. Twice differentiable continuous function, and we are not able to in general find out the solution. In many situations, where the operator is not engineer we are not able to find the solution so we want to discretize and solve this problem numerically and not analytically.

So we have created this grid points we have marked the dependent variable value at grid points as ui and then we converted this boundary value problem the differential equation first we converted into a set of algebraic equations.

(Refer Slide Time: 03:46)



So that was done by Psi. So the approximated the second derivative is in Taylor series we approximated the first derivative using Taylor series. So at all the internal grid points, see here these are internal grid points z2, z3, z4 up to zn these are all internal grid points. And then z1 and zn+1 are 2 boundary points. Okay. Z1+zn+1 are 2 boundary points whereas—so at all the internal grid points we enforce this equation, okay. Please note that enforce in this equation does not mean we are solving the exact problem.

We are solving an approximate problem and for the approximate solution we enforce this =0. There is an alternate way of solving this problem is to create to hypothetical grid points one on the either side of the one on the either side of the endpoints and then discretize either way is fine if delta z is small. So for the time being let us write the equation in which we use one sided derivative approximations.

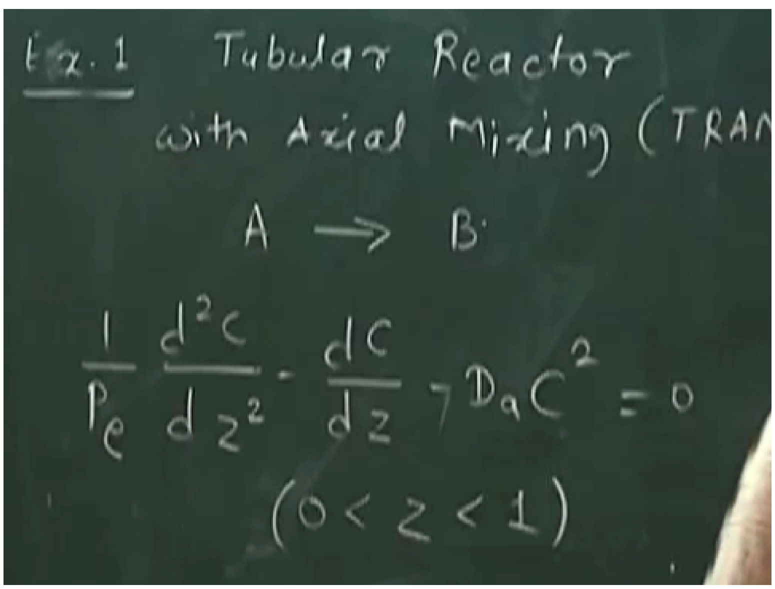
So this what I get here is the approximation, this approximation is nothing but in the notation that we used earlier, it is nothing but y cap = or y tilde = T cap x tilde. So we started with some problem, we discretized and we got this discretized problem. So we started with a differential equation and 2 boundary conditions. We got set of algebraic equations. So a transform problem is completely different from the original problem.

We hope that if you choose delta z small then the Taylor series approximation is a good approximation and then this transformation will give you a solution which is close to the true solution. This will not—this way of transforming will not recover the original solution. You can only recover approximate solution. If you reduce delta z then you will get better and better solution but the price that you have to pay is that smaller the delta z more the number of equations. And then you will realize that these equations are coupled.

For ith equation requires u value of i+1 and i-1. So these equations are coupled. So if you let us say take 100 internal points, delta z is very, very small if I take 100 internal points, okay. Then I will have hundred equations here and 2 additional equations 102 equations and 102 unknowns to be solved simultaneously. How do you solve this? Newton-Raphson method. Or if this turn out to be linear equations solve them using Ax=B simple Gaussian elimination.

So you are transforming the problem and just solving it. Let us look at some examples today. So this is the background and we catch up from here.

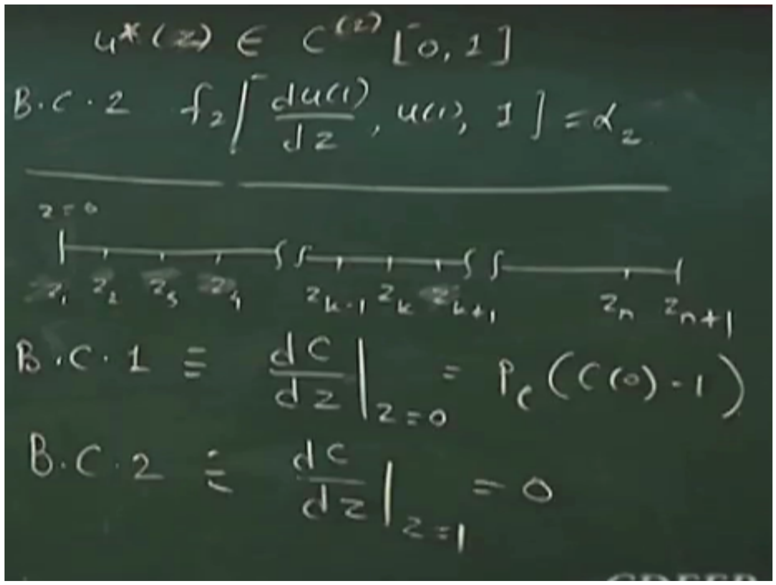
(Refer Slide Time: 07:25)



So my first example is going to be a Tubular Reactor example. So my first example is TRAM or Tubular Reactor with Actual Mixture. I am going to just take the equations and then we will see how to discretize, that is the key thing here. So this is the simple reaction where A goes to B, okay. A very, very simple reaction where A goes to B. And the ODE BVP that is given to you is 1/Peclet number * d2c/dc square/dc/dz – this is Damkohler number * c square = 0.

So this should be obeyed between 0 z 1. Okay. This is my differential equation. And then I have 2 boundary conditions. I have 2 boundary conditions. So my boundary conditions here

(Refer Slide Time: 08:47)

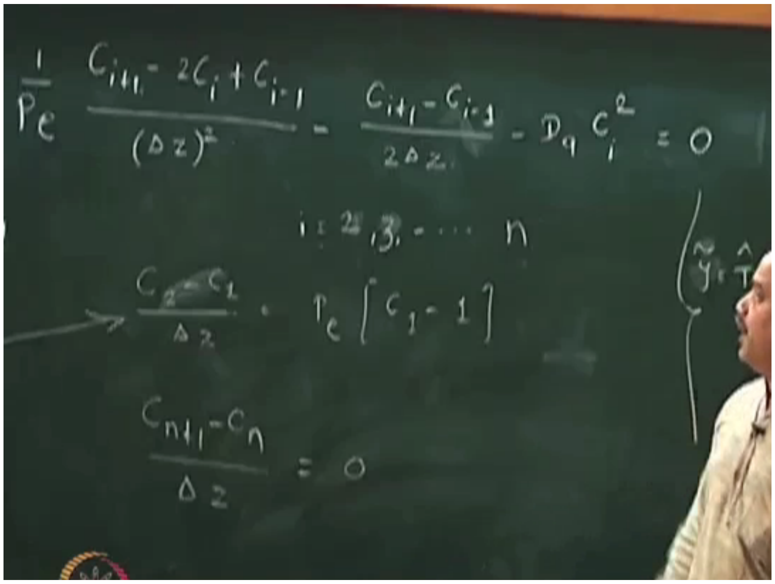


So my first boundary condition at z = 0 the derivate at z = 0 the derivative at z=0 is peculiar number * C concentration at 0-1. And my second boundary condition at, so this is my this is my second boundary condition, I am just going to apply the method of finite difference, this that we are developing is called as finite difference method because we have developed approximation of first and second order derivatives using Taylor series approximation, we have developed finite difference approximations. Okay.

So now first task, what is this operator Psi here. This operator Psi here is nothing but this differential equation. Okay. So this second order derivative I am going to replace by its approximation at any ith point okay. Same thing is true about this derivative I am going to replace approximation of this derivative at any ith point and so I am going to force the residual the term that you get is called is residual, the residual is force to 0 at all the internal grid points at a 2 boundary points, I am going to use 2 boundary conditions.

So this equation will now be replaced.

(Refer Slide Time: 10:12)



By 1/peculate number, so this will be ci+1 – 2ci + ci-1. Now what are the ci, C1, C2, C3? C1, C2, C3 are concentrations at the grid points. So at this point the concentration is 1 at this point, 2, C3 at in general at this point is ck, ck+1 and so on, cn and cn+1. So these are the dependent variable values at the grid points. Okay. At any ith location I am discretizing the differential equation. So the first derivative at ith location, the first that is second derivative is replaced by this.

Then –ci+1-ci-1 divided by delta z, divided by 2 delta z. Then what is the next term, Da ci square = 0, this algebraic equation, this converted algebraic equation should hold at all the internal grid points i=2; i=3 up to i=-n. Okay. So suppose I create let us say total grid points is 100 so 98 internal grid points 2 boundary points, okay. Then I will get 98 equations like this. Okay. Each equation is re-relate to the neighboring values of c. Okay.

How do I discretize this? What is my first boundary condition? My first boundary condition is, this will be C2-C1/delta z = peculate number * C1-1, so this is one more equation. I have just converted this into a difference equation finite difference method, all right. I have converted the first boundary condition into a difference equation. Now I will convert the second boundary condition into difference equation.

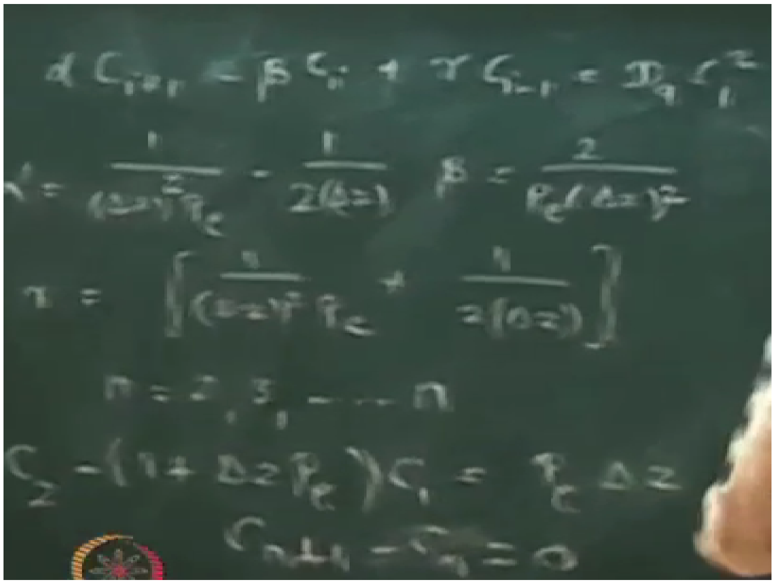
So this will be cn+1-cn/delta z = 0. How many equations I have now? Not 3. N+1 equations. How many unknowns I have? N+1 unknowns. If I solve this n+1 equation and n+1 unknowns I will reconstruct a solution an approximate solution how the boundary value problem, okay. It is difficult to solve analytically because of this ci square, and in general I have taken very, very simple reaction, okay.

It could be a 2a so 2a going to be or something like that. So I have taken away a very simple reaction. This could be a more complex equation here. Okay. So this n+1 equation in n+1 unknowns I have to solve them simultaneously to. Now I want to point out something else here in addition to talking about discretization, I want to talk about some nice structure that appears. Okay. I am going to rearrange this set of equations.

And then write them in a specific form to give you insight about something that we will be talking little bit later. I am going to talk about a Sparse system. Now this particular equation, this is a very nice equation if you look at it, there are n+1 variables but only 3 variables appear in one equation. Only 3 variables appear. Okay. And when you have large number of equations you can actually make use of this fact that there are only 3 variables are appearing in each equation. These kind of equations will give rise to what I called as Sparse system. Okay.

So we will side by side as side note I will also introduce this idea of sparse system and then later on we will be looking at special algorithms to deal with these sparse systems. But right now it just to give you an idea which is. So this first equation and this together I am going to rewrite in a matrix form which will give me a sparse equation. Okay, so if I collect all the terms of you know, ci, ci+1 and ci-1 together then I can write this equation as:

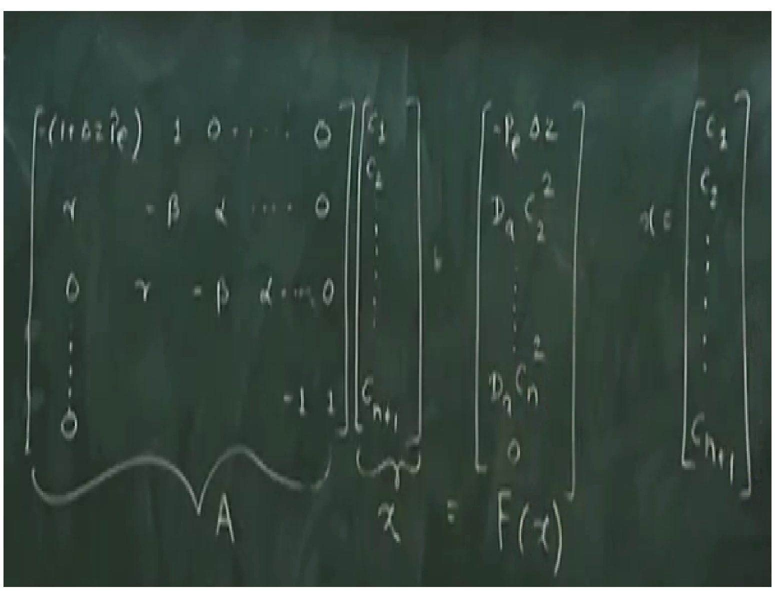
(Refer Slide Time: 16:02)



Alpha ci+1 – beta ci + gamma ci-1 = Da ci square. Okay, where alpha is 1/delta z square. Okay. I get equation like this n=2,3 up to n. I have just grouped the terms together. And then we have 2 more equations. We have 2 more equations coming from the boundary conditions. I am going to combine this and I am going to write this as a matrix equation. So if I include the 2 additional boundary conditions okay. What are the boundary conditions?

One boundary condition will give me this equation. The other boundary condition will give me cn+1 – cn = 0. So this I have n+1 equation and n + unknowns

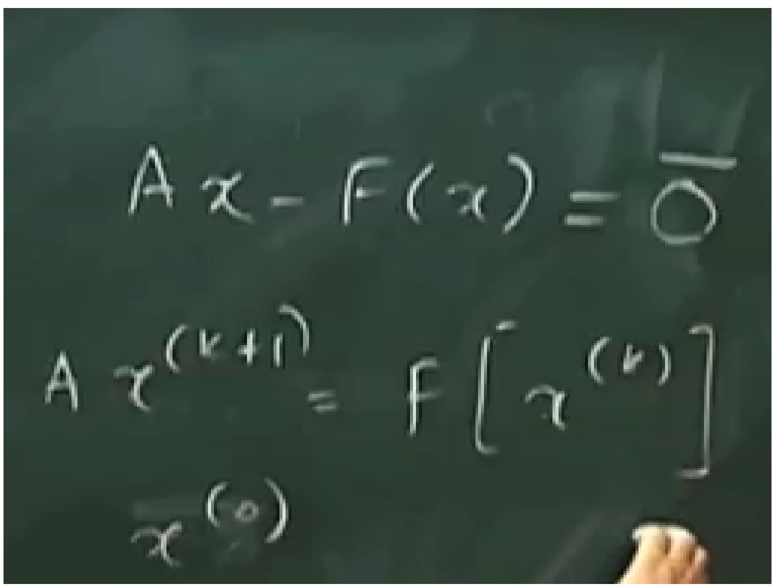
(Refer Slide Time: 17:43)



I am going to write them in this particular form. So this particular equation will be set of n non-linear equation in n unknowns. **"Professor to student conversation starts"** You might be wondering initially when I started talking about Newton–Raphson method in multiple multivariable domain; where you get this? Where do you that you get this equation, classic example. Okay. **"Professor to student conversation ends"**

One boundary value problem is giving me if I take 100 grid points I will get 100 equations. Okay. So large number of equations. This equation to. Okay. See this is like matrix A operating on vector x which is f of x. We have to solve Ax=f of x. What is x here? X here is this vector C1, C2, C3 and Cn+1. Okay. In abstract form, this equation, one matrix equation is nothing but matrix A operating on x gives me f of x. I have to solve this problem, or in other words, I have to solve Ax – f x = 0.

(Refer Slide Time: 19:29)



You can solve this by Newton-Raphson. You can solve this by simple iterative methods, for example I could solve this by a simple method which is iterative method of this form Ax k+1 = f of x k. Okay. I start with a guess X naught okay, so I start with the guess of concentration profile, what is x, do not forget that. This is concentration profile. This is where your input as a chemical engineer will come to picture. You have to give a sensible concentration profile.

If you suppose give here negative numbers, it is not going to work. So during initial guess somebody has asked, is it very important? It is very, very important, how do you initial guess. Okay. So this is my concentration profile discretized concentration profile and I can give a guess if I put this guess x naught here I can generate the new guess by solving Ax = f x naught. Then x1 I can substitute get x2, x2 I can substitute will get x3 and so on.

And then I can see whether the difference between this and this is going to 0 or not by checking norm of this and see whether algorithm converges. Okay. This is one way of solving. Right now that is not important. What is important is that if I solve it like this, this A I have to solve Ax=B, so you can one side put X0 here okay. This f of x is known value, okay, I have to solve for x1 by solving the linear system Ax1= okay.

Then I have to solve another linear system Ax2=fx1 and so on. So I have to solve large number linear systems, linear algebraic equation. Now if I come back here, if I come back here okay and then if I look at this matrix there is something special about this matrix. **"Professor to student conversation starts"** This is a? Some of 3 diagonal but upper one is not diagonal. So this is a matrix, this is called as tridiagonal matrix. It is a banded matrix, there are lot of zeroes.

How many elements is matrix has? Let us say if I take if I take 100 grid points. How many elements this will have? 100+100, if I take 100 grid points, this will have 1000+1000. And we said more the number of grid points better is approximation. So I am force to take smaller delta z, if I force to take smaller delta z more number of equations, matrix will have large number of elements. Okay. **"Professor to student conversation ends"**

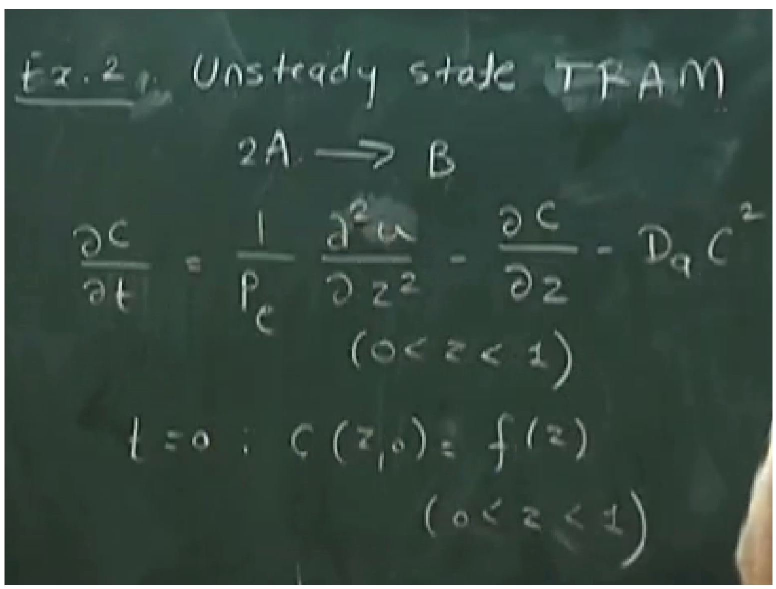
If I develop some special method that integral with these bended matrices, then this iterative calculation will become very easy. Okay. So later on we are going to look at these methods for sparse system, this is a sparse. So we will develop special method for dealing with sparse linear systems which will reduce the computation. Because, suppose you need 200 iterations 200 times if you have to invert a 1000 X 1000 matrix not a nice thing to do. Okay.

Instead if you if you develop a method which takes into account there are lots of 0s okay, you can able to do calculation very fast, that is where all these sparse matrix methods come into the picture. There is a side note we will need this very often and when you do the discretization using finite difference you will see that almost every time you hit into this sparse matrices, sparse matrix. Okay. Some form of the other. Okay let us move onto some other example.

But it is not necessary that this finite difference method can be used to solve only boundary value problems. It can be used to construct approximation solution of partial differential equation as well. Okay. I will take the same TRAM problem and show that how you can convert this problem, a partial differential equation. Right now I took the steady state problem, I will look at the problem which is time varying.

Now in this case we got algebraic equations. In the partial differential equation, we might end up with something else. Okay. So what is it that will end up with?

(Refer Slide Time: 24:08)

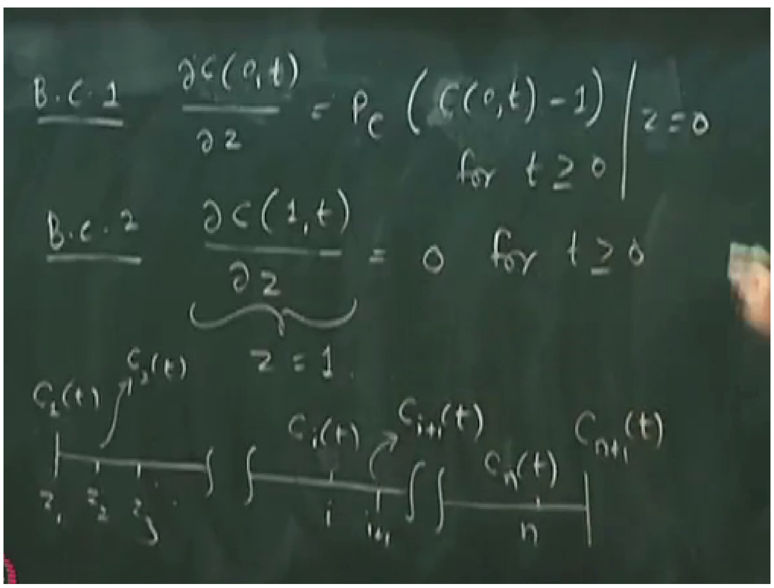


Now my second example is a PDE. Say same unsteady state, same tabular reactor with axial mixing. Same TRAM, okay with except that I am going to consider the unsteady state condition not the steady state condition. Okay. So here, so let us say this is the reaction in which 2A goes to B and then we have this dou c/ dou t that is – rate of change of concentration insight the reactor okay. This is 1/Peclet number.

Now unlike the previous case where I was looking at the steady state behavior I am looking at a transient behavior, I am looking at a unsteady state behavior. So dou c/dou t is not 0. Okay. The concentration is a function of time and space not just space. Earlier we looked at the boundary value problem that came that arises when you look at a steady state of behavior of this particular system. Okay, I still have the same 2 boundary conditions.

But I will also have an initial condition now. Okay. There are 2 boundary conditions at z=0, z=1 and is also initial condition in time. Okay. So a t=0, I have some concentration profile I have some initial concentration profile okay, inside this is given by fz okay. This is my initial condition there is some concentration profile inside and then I have 2 boundary conditions.

(Refer Slide Time: 26:00)



Sorry this is not 1 this is t. So this boundary condition is at z=1, this boundary condition is at z=0, so the solution now, solution now is actually function of 2 independent variables, one is time one is space and other is time. At z=0 for all time this boundary condition will hold at z=1 at z=1 at all time this boundary holds, okay. I want to know discretize this particular problem using finite differential method. Okay. Using finite difference method.

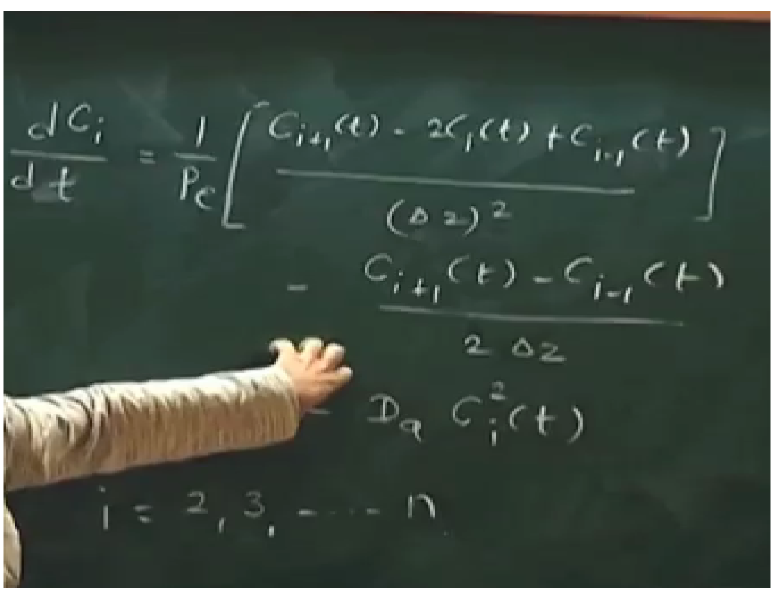
What I am going to do is, I am going to discretize the space and leave the time untouched for the time being. How will I solve the resulting problem? Something that we will see later, but we will convert into standard form which can be solved using a standard tool. Okay. So in this case this partial differential equation, I am going to convert into a set of ordinary differential equation. Okay. I am going to discretize in space not discretize in time. Okay.

So what I will get is a set of differential algebraic equations. So how will I discretize this? Okay the same trick that we did earlier is space, we are going to you know we are going to denote this grid points z=1; z=2; z=3 and then you have this C1, okay. Again I have discretized, I have discretized my domain except now the dependent variables c that is concentration at the grid points is not only function of space is also function of time. Okay.

I have discretized only space so C1, so the discretization in space appears through this indices 1,2,3,4,5 up to n+1. Each of them is a continuous each one of them is a continuous function of time so I am going to only discretize in space. Okay. So now what will be the residuals? I write I enforce the partial differential equation only at the grid points. Actually the original partial differential equation holds at every point inside a domain at every time.

They are not able to do that because will get large number of equations. Okay. If you start infinite number of equation if you start forcing at every point, you have to discretize and. Okay. So now if I discretize this I will get dCi/dt.

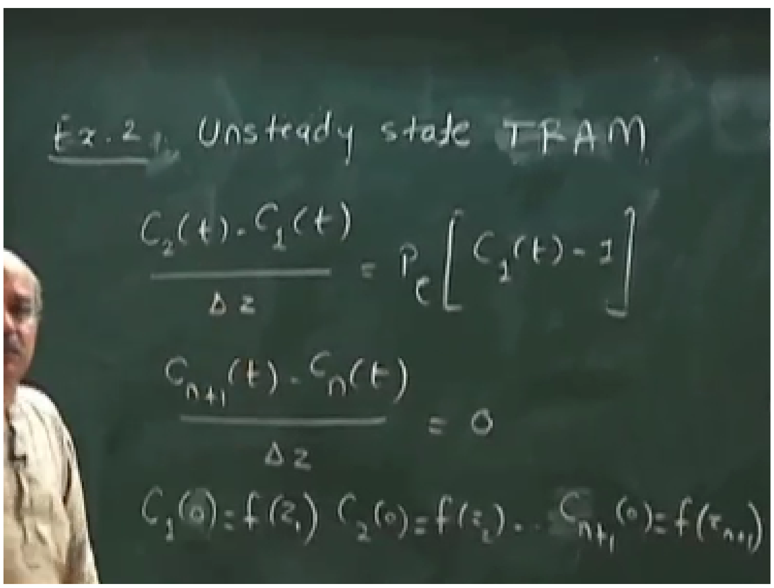
(Refer Slide Time: 29:03)



See the differential equation is forced at the ith grid point okay. What will be this? 1/Peclet number * Ci+1 t – 2Ci function of time. Well –Ci+1 t - Ci-1 t/2 delta z. What I have got here, is set of ordinary differential equations. I started with the partial differential equations. I started with one partial differential equation; one partial differential equation got converted into large number of ordinary differential equations. Okay.

So in this case the original operator T is partial differential equation that transform the operator is set of algebraic and differential equation. Why I am saying algebraic? Because I have 2 more conditions here. Okay. One of the, well in this particular case you will get 2 more differential equations. What are the 2 additional differential equations? The 2 additional differential equation that you get here are: Well, this is a derivate in space, you will get algebraic equation here not the differential equation.

(Refer Slide Time: 30:43)



So what will be here. C2 t-C1 t/delta z. I am taking forward difference approximation here. And =peculate number into C1 t -1, so this is one algebraic equation. Okay. What is the second algebraic equation? So what I get here is the differential algebraic system. Okay. I got differential equations in time and I got 2 algebraic equations. Actually this is a simple problem in this particular problem I can eliminate 2 variables.

And I can convert this problem into n-1 ordinary differential equations. I can treat it like that or alternatively I can treat those ordinary differential equations and these 2 algebraic equations together into a single DAE Differential Algebraic System, towards the end of the course we will be looking at how to solve differential algebraic system. So there are 2 ways to go from here, one is to use these 2 constraint eliminate C1 and eliminate Cn+1.

You can get equations only in terms of C2 to Cn. N-1 differential equation in N-1 unknowns. And then, well you cannot stop here, you also have to discretize initial condition, this is a differential equation it will need initial condition, so you cannot just stop at here discretizing the boundary conditions. What about the initial condition? What will be the initial condition? So initial condition will be C1 0 = f z1 C2 0 = f z2 and Cn+1 0 = fn+1.

So these are the initial conditions. These 2 algebraic constraints coming out of the boundary conditions and then n-1 ordinary differential equations have to be solved simultaneously. Okay. Have to be solved simultaneously So we have transformed the problem which was the partial differential equation which is difficult to solve analytically because you have Ca square term appearing. Okay. Analytical solution could be difficult in this particular case.

In general, if you have more complex reaction rate equation it will be very difficult to solve this problem analytically, you have to solve it numerically. Okay. So you have to convert this problem into set of ordinary differential equation, initial value problem, or differential algebraic equation with initial value specified. Okay. So these equations have to be solved simultaneously. The transform problem is different from the original problem.

So remember this, let us look at some other partial differential equations. Okay. **"Professor - student conversation starts"** Is the idea clear, what is happening here? Here we transform a partial differential equation into a set of ordinary differential equation. Earlier a boundary value problem was transformed into set of algebraic equation which were non-linear which has to be solved simultaneously.

Remember this set of ordinary differential equations is not linear ordinary differential equations. This has to be solved numerically using some method like Euler method, Runge–Kutta method or whatever, we will be developing these methods later on. Right now I am just worried about problem transformed. Okay. I am only worried about problem transformation. Yeah, no, I have n+1 variables and I have n+ 1 equation, out of this n-1 equations are differential 2 are algebraic. Okay. **"Professor - student conversation ends"**

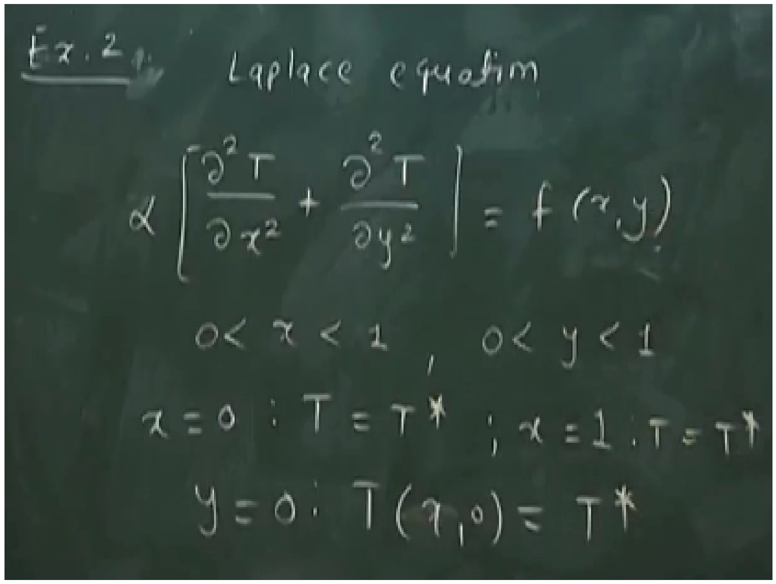
So I can eliminate 2 variables from this ordinary differential equation, using the 2 algebraic constraints and convert into n-1 ODE initial value problem. That is possible. In this particular case it is possible. If your boundary conditions had some non-linearality it is not possible to convert into very easily into ordinary differential equation, where it will be a DAE system it has to be solved as DAE system. Okay.

My third example is going to be a partial differential equation in 2 dimensions, okay. So 2 special dimensions. So this is model of a Farness, okay, temperature distribution in a farness. The 3 walls of the farness are insulated and a constant temperature idealization, okay. And then there is a you know convective heat transfer from one of the phases. I mean if you ask me how do I model this particular room, temperature distribution in this room, I would use equation something like this.

Well, why is this type of farness equation? Well you can see that this phase is not insulated okay. You could say that from this phase that convectively transferred to outside, these 3 walls let us assume as an idealization are insulated perfectly at a constant temperature, where is the heat being generate? Well, each one of you is like a bulb of 40 watts, so a 40 students * 40 watts, there is so much heat being generated inside this room, okay.

So the temperature inside this room is function of 2 variables x and y. Now where the heat is distributed, the heat sources are distributed, all of you are sitting along different places okay. So what is this model?

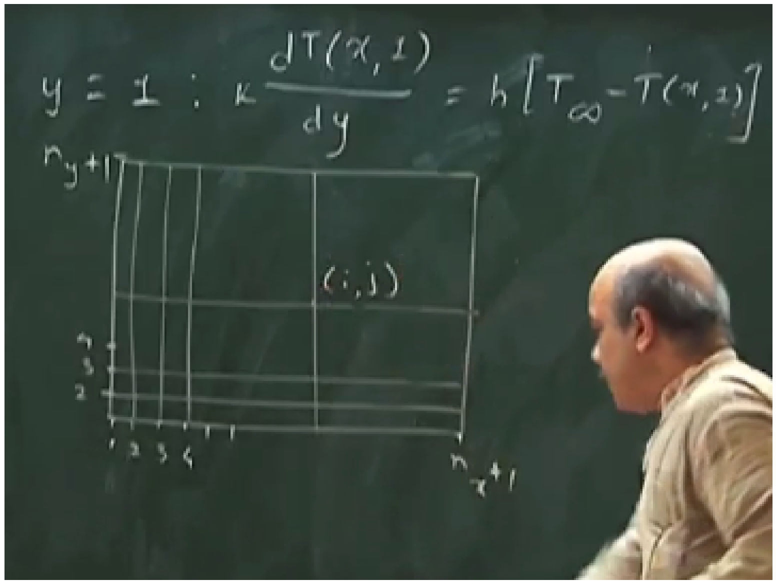
(Refer Slide Time: 36:59)



So this is the famous Laplace equation. Okay let us initially consider the steady state problem. All of you are perfectly generating the same amount of heat okay at all time, there is perfect steady state. Okay. Now there will be 4 boundary conditions here at 4 different boundaries. So we said that the 3 boundary conditions are—so this equation should hold between 0, x, 1, I normalize the lens between 0 and 1.

So this partial differential equation should hold at every point inside this room okay. And then I have 4 conditions. My 4 conditions are x=0; my T=T* insulated boundary. Okay. Then x=1; T=T*. Okay. When I say T here which means T along the boundary. Okay. I am taking shortcuts and not writing --. Then at y=0 I have T=-- so I should write T x 0=T* so accordingly I should write even these equations. Okay. So 3 boundaries have constant temperature and the 4th boundary.

(Refer Slide Time: 38:45)



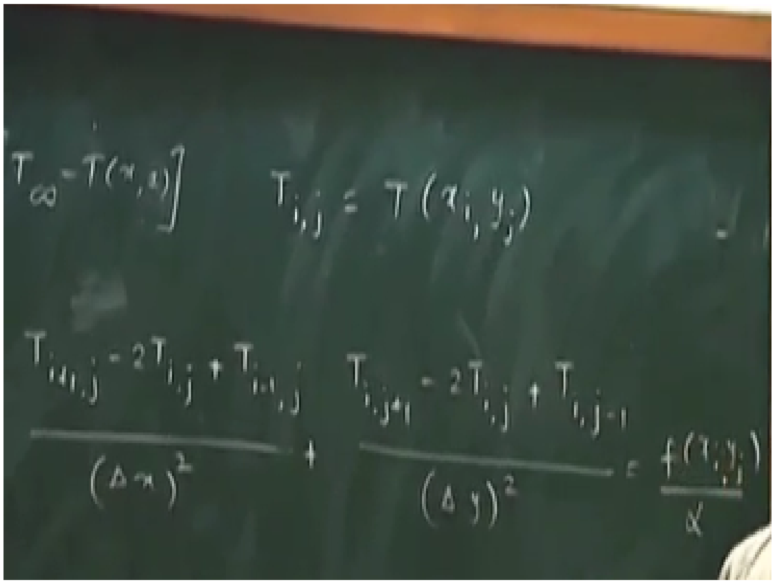
So I have convectively transferred at the 4th boundary at y=1 I have k. Okay. So I have 4 boundary conditions here. I have 4 boundary conditions here. Okay. Now how may I be going to discretize this? I am going to use Taylor series approximation in one dimension at a time. So I am going to I am going to discretize this domain okay by constructing grid points, now my domain is something like this.

I am going to construct grid points here, so this is my 1,2,3,4 and so on. This is my okay. So finally I am going to get nx+1 grid points along x direction. And let us say ny+1 grid point along y direction, 1 to ny+1; 1 to nx+1. So if you draw these lines here parallel to the—so in general in general I am concerned about forcing the differential equation at some iz point. Okay. I am constructing the grid here and discretizing.

See earlier we had only one spatial dimension, discretize in only one spatial dimension. Now I am going to discretize in 2 spatial dimensions x and y. Okay. And these partial derivatives are going to be approximated in 2 spatial dimensions. So my partial differential equation. **"Professor - student conversation starts"** Where should this partial differential equation will hold? Everywhere. Actually it should like all the points. Okay. Even for one dimension we created 100 grids points.

Suppose you create 100 like this and 100 like this, how many internal grids points will be there? 100 X 100 okay. These are 100 X 100 internal grid points and you want to force the partial differential equation at every grid point. Right, okay. **"Professor - student conversation ends"** So in general at I, j point I can write this equation as

(Refer Slide Time: 41:38)



Well we are going to use this notation Ti,j = T xi, yj, so temperature, this is to simplify the discretization process, I am going to use Ti,j is temperature at xi,yj okay. So this is ith vertical line and this is jth parallel line. Okay. At this point, so I am going to write this equation and so what will be the—the first one will be Ti+1j – 2 Ti,j + Ti-1,j/delta x square + Ti,j+1, see I am taking partial derivative in x direction, partial derivative in y direction 2Ti,j + 2.

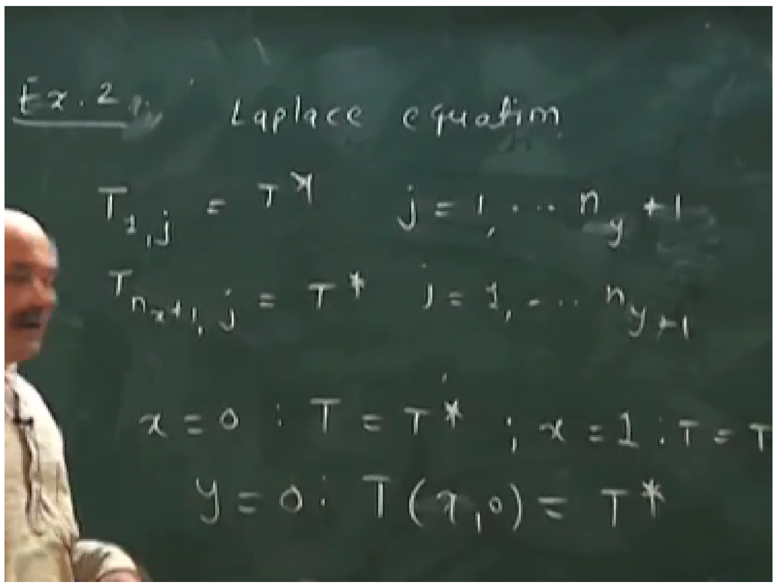
"Professor - student conversation starts" How many such equation are we get? I am going from 2, 3 to nx, j going from 2,3 to ny. How many algebraic equations I get from a one partial differential equation? I get nx-1*ny-1, right. Where are the additional equation coming from? Boundary condition? Boundary condition. So I should discretize the boundary conditions and then take all these equations together okay.

How many equations and how many unknowns? nx+1 X nx+1, okay. 100 X 100. If you take 100 grid points. Very modes requirement. Just imagine this room 100 grid points, not to in, if you want a better solution from me it should go 1000 1000. But the number of variables will be very,

very large. Okay. Yeah? (()) (44:31) I am not get into a physic related problem. Even this problem you going to solve it. Let us—So now we will require additional equation right, at the boundaries. So what are the boundary equations? (()) (44:46)

No, no, no. Let us not get into physics. You do at in other course, transfer course, you ask that question. I am just worried right now about problem discretization. Okay. So now, so this will be addition equations. **"Professor - student conversation ends"**

(Refer Slide Time: 45:06)



So I will get T* at x=0; x=0 I will get all these points, right along y. So I will get T1,j = t* j going from 1 to ny+1. Then the second boundary condition will be T nx+1,j = T* j going from 1 to ny+1. Okay. Then you discretize the third boundary condition. Okay. And then this 4th boundary condition you discretize using forward difference-- sorry backward difference. You discretize using backward difference. Okay. At all the grid points along x, okay.

"Professor - student conversation starts" Yeah? (()) (46:18) Mixed derivative? (()) (46:22) You do not get second order derivative of the boundary. (()) (46:29) Yeah, so why do not we develop an approximation, a difference equation? Can we develop a difference equation so you take first derivative in x then you take derivative in y?" So you will get large number of equations which are coupled in general and they have to be—this will also give you a sparse system you can see that. Any equation here—

How many variables appear? (()) (46:58) Only neighboring 4 variables appear. If you take Ti, j, i, jth point there are only 5 variables appearing. How many variables we have? We have large number of variables, nx+1*ny+1. But in one equation there are only 5 variables. You can expect that if you try to solve this numerically you will get sparse system, okay. You get a sparse system and then of course we will be looking at solving sparse linear system or solving a sparse non-linear system separately. But the—is the idea clear?

How do you do discretization? Okay. **"Professor - student conversation ends"** So in this case the original problem is a partial differential equation, discretized problem is a set of non-linear or linear algebraic equations which have to be solved simultaneously. So the transform problem is different, the solution that you get from the transform problem is an approximate solution, okay not the original solution. So in the next class, we will see one more example one more way of discretizing this and then we will start with some other methods of discretizing.