NPTEL
NATIONAL PROGRAMME ON
TECHNOLOGY ENHANCED LEARNING

IIT BOMBAY

CDEEP
IIT BOMBAY

Phase field modeling; the materials science, mathematics and computational aspects

Prof. M P Gururajan
Department of Metallurgical Engineering
And materials Science, IIT Bombay

Module No.9
Lecture No.38
Diffusion equation:
Imposed concentration BC
& explicit/implicit methods

Welcome we are looking at a numerical solution to the diffusion equation, we look at a boundary condition in which it the composition was at some value on the left side and the flux was zero at the right end so that is basically directly and Neumann boundary condition so in this lecture I want to talk about two boundary conditions one is if you put directly on both ends what happens okay, so that is in other words we are looking at a system like this so this is a one-dimensional system like I said, so there is diffusion barrier on this side on all sides.

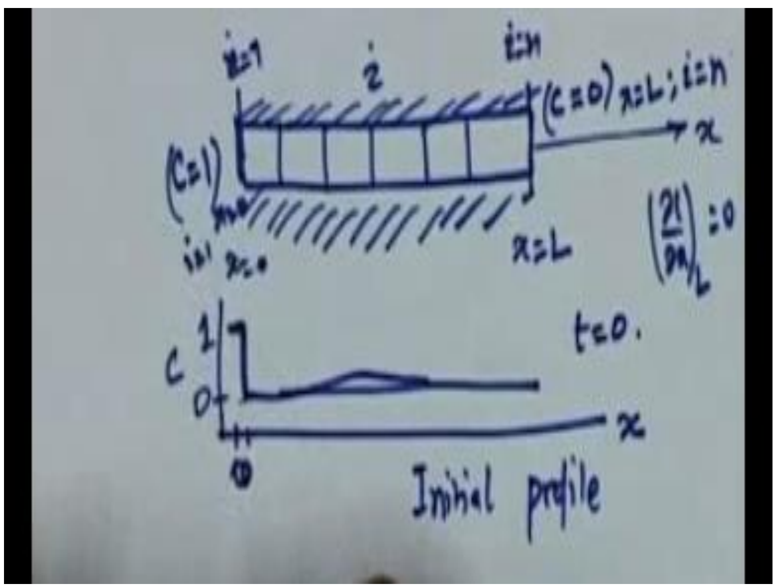
Okay so you think of it like a pipe and then on all sides it is there so this is the x-axis right so this is the x axis and this is x = 0 this is x = L and we split it in two nodes, so this is n equal to this is i= 1 and i becomes equal to n here okay, so we split it into several nodes at the at the different nodes this is the i2 node, is what we are looking at then the boundary condition that we implemented yesterday had c = 1 at this point at x= 0 or i =1.

And we had this boundary condition namely δc/δx at L =0 that we saw yesterday but today I want to put the other boundary condition i want to make c =0 at x = L that is the same as saying i

=n and, I want to see what happens to the composition profile? Initial profile is that it is 1 here and then it is 0 everywhere right, this is the initial composition profile okay so this is 0 this is 0 for the x and this is 1.

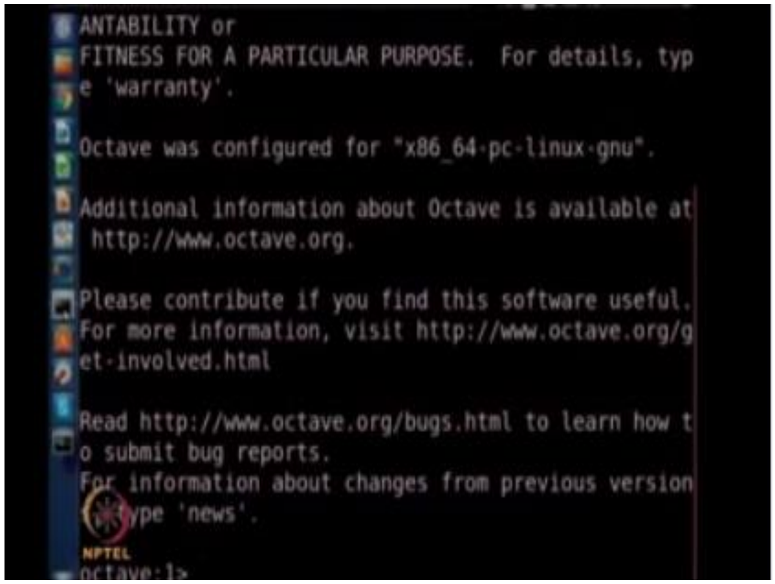
Okay so we want to know what happens so this is 0 for this is this is 0 for the position, so there it is 1 and we want to know what happens to this initial composition so this is at time t = 0 so, this is basically the initial profile. So if you compare with what we did in the last lecture everything is the same except that this boundary condition is different of course you know how to deal with this boundary condition because we have already dealt with this kind of boundary condition earlier okay.

(Refer Slide Time: 02:53)



So I am going to write explicit and implicit the code for doing this and we are going to write it in script mode and we are basically going to follow whatever we did yesterday except for this small change everything is going to remain the same so, of course the way I do it is as usual we go to the computer we open a terminal we go to the relevant directory this is very important and this is where I will invoke okay.

(Refer Slide Time: 3:27)



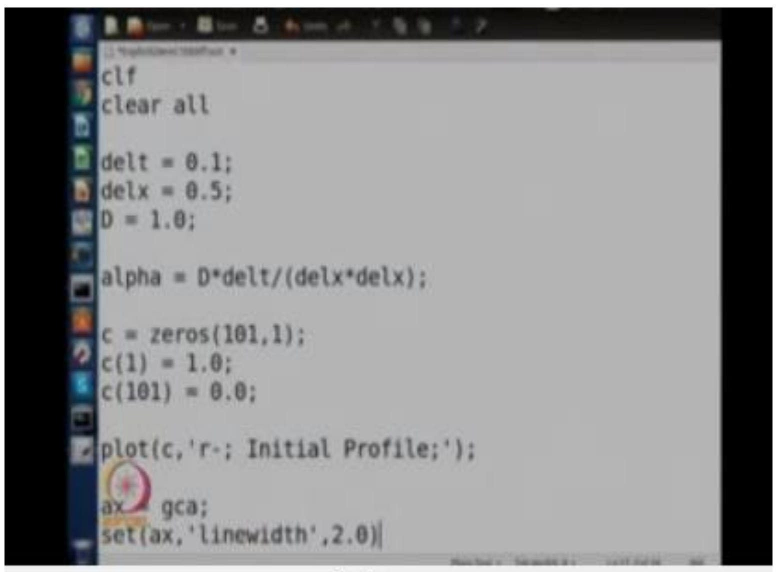
Now to run in octave I need a script for that I use the g edit to generate such a script, I'm so I am going to save this as save us let's go and save it in the same directory where we want to run it and let us call it as explicit right, 0 C 1 D diffusion auto city right so, we are looking at explicit but 0 composition at the right hand and1d diffusion equation so I call it as explicit 0 C 1 D diff Tata city.

As you can see when I name the files i use capitals and small letters because operating systems like Ubuntu distinguish between capitals and small letters it is a good idea when you are using a Linux operating system not to leave spaces in the file names are dashes so either you use underscore or you can mix and match capitals and small letters to make it easier for you to read the file name in okay if you have capitals instead of living space then it helps you break and read the file name properly so i have named it as explicit 01 d dot OCT.

Let us save this file and as usual we are going to start with first clear all the figure then clear all the variables that it has in its memory, let us be fine del T as some 0.1 okay and then del x as some 0.5and we are going to define D as 1 so which helps us define alpha as D*del t/(del x*del x) right so, we have alpha so I am going to define the initial profile c = zeros(101,1) and c(1) = 1, c(101) = 0.0 right, so we need to fix the values we need to fixit at the two boundaries one boundary is always at 1 the other boundary is always at 0 so that is what we have done.

Okay so we want to plot the initial profile plot c and we generally want to plot it in a different color and we also want to say that this is the initial profile, so that itself okay so, these Colons semicolons and upper strophes everything you had to be very careful if you miss any of them you will start getting error messages okay, typically we also do some cleaning up of the plot at this point so we say ax = gca it so this is to get the handle and then we say set ax, line width that is 2.0 sorry.

(Refer Slide Time: 07:08)



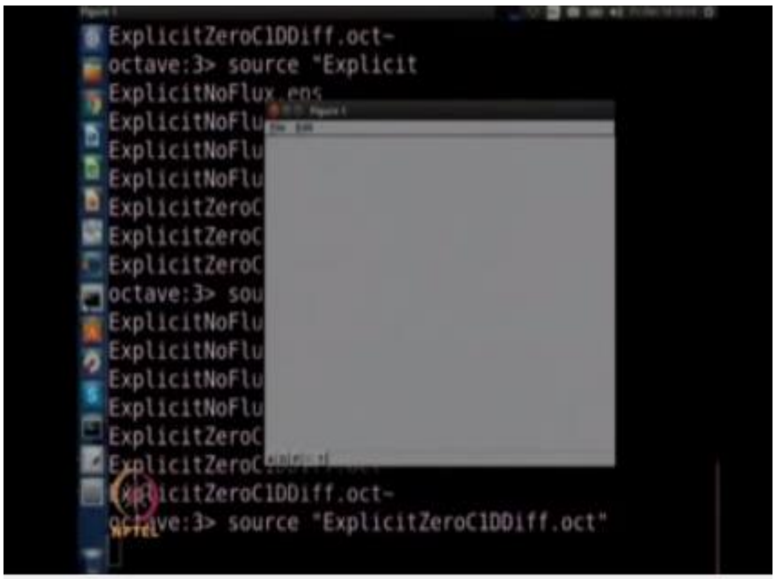
You can take these scripts and play around with these numbers to see what happens I mean that will give you a better idea of what these things are doing ok so axis is square, of course there are ways of finding out what these commands are once you know what the commands are because for example you can go to the octave window and look for help access right so it gives you all the information about this command called access right, so help gca ok so it gives all help for gca and so on and so forth okay.

So there is a way to get the more information on this so I am just introducing you to these commands so that you can go look it up and figure out more of course after this we always say hold on so that the figure will be saved. What do we do we introduce three loops one is for plotting at different times the other one is for evolving for some given amount of time and the third one because we are doing explicit we need to do but now it is sufficient.

If you go from 2 to 200 because for 1 and101 we have already given the boundary condition and the composition is not going to change for all time at those points so we no need to worry and what is the expression c(i) is nothing but c(i) this is all now the old c(i) which will be stored in the new c(i) on the left-hand side into (1 - 2 * alpha) + alpha *(c(i-1)+ c(i+1); right this is the expression we have so c * into (1 - 2 * alpha) + alpha *(c(i-1)+ c(i+1).

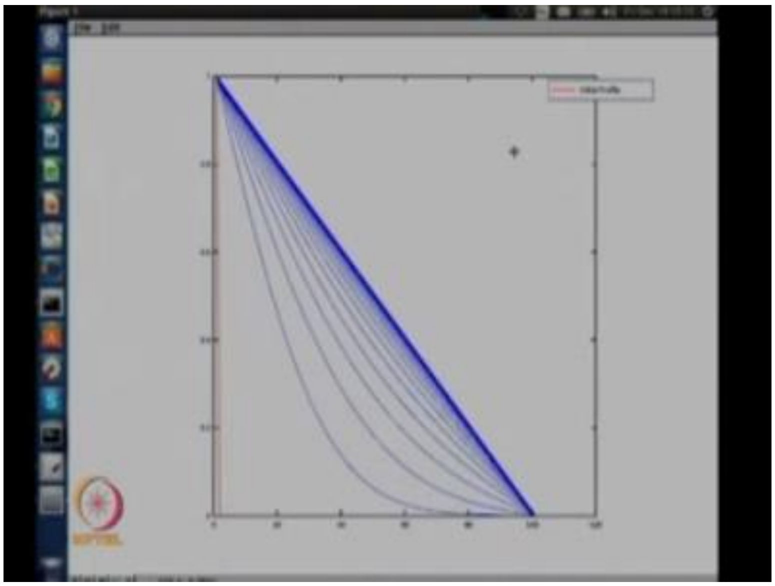
So we end this for loop we end the for loop further times and we then do the plotting of c and we end they last for loop and typically we also print the resultant files, so let me print to the device we can print to eps color for example and we need to give a name explicit and i will name it as zero c.eps, 0c indicating that the composition is kept at zero value on the right end. Let me save this file and let me go and run it.

(Refer Slide Time: 10:23)



So we name this file as explicit and if I press tab it tells me what are all the files which start with the name explicit and then 0 there is only one file dot Oct. okay, so let me source and then I get the screen which means that the figure is cleared and it is ready and it is going to then plot the initial profile and the profile at different times and that is what we are going to see, so here is the figure that has come see we used line with 2 so these box things are a little bit thicker, if not they will also be as thin as these lines.

(Refer Slide Time: 10:48)



So sometimes it is a good idea to make these boxes or thicker okay and I don't think so if you look at the initial profile yeah we plotted it as a line so you see a red line and below this you do not see because it is just smack on this axis. And then the composition profile is setup and as time goes by then a steady state is reached where the material comes here and then it leaves here because it has to maintain this composition at z =0.

So depending on what is the amount of material that is coming in at any point you know the slope of these curves with that slope, it'll leave from here so there is a flux at this point that is how it is able to maintain the composition at 0 for all times, which means if you have some small flux here that flux will leave if you have some larger flux it will leave and then as the flux becomes larger and larger.

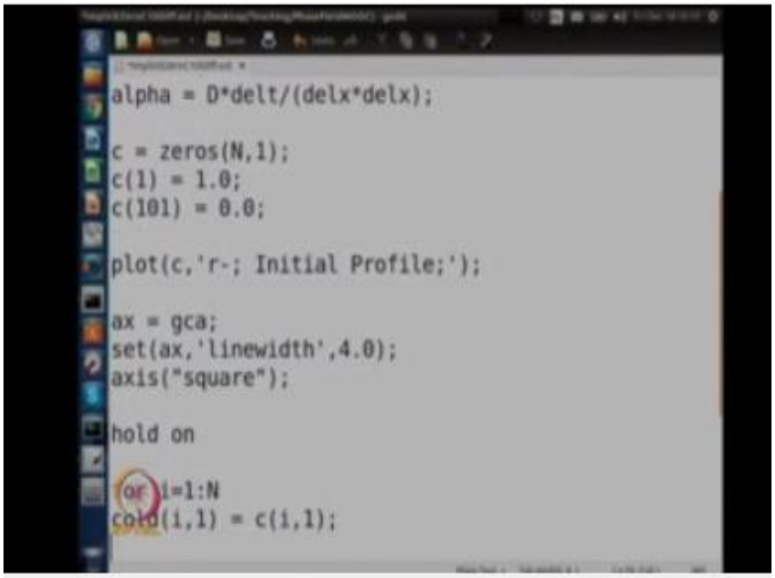
So finally there is the same constant flux everywhere so it becomes a straight line everywhere, so this solution is known and you can obtain this solution by just looking at the diffusion equation and you find out when $\delta c/t = 0$, so then it becomes the Laplace in and $\delta^2 c = 0$ or if you keep the D inside it is $\delta/\delta x(d\ \delta c/\delta x)=0$ which means $d\ \delta c/\delta x$ is equal to a constant and that is the solution we are getting, because after a long time I mean it sets up a profile which is a steady-state profile.

Okay so this is using explicit scheme of course explicit scheme is always preferred because we find that in the explicit method it is just an algebraic expression, of course we can do this implicitly like we have done yesterday for the other problem. So let us try to write the script which will do the same thing implicitly, so for implicit now I am going to use the same file and I am going to only make some modifications.

Okay so I would like you to actually copy the file that you have or keep the same thing because you can always save it again so let me save this file save as now let me make it as implicit zero concentration 1d diffusion dot Oct so that is how i have saved, okay the first step of course we are going to clear all the figures clear all the data delta del t, del x, D alpha, you have to define the initial profile you have to put the boundary conditions, you have to plot the initial profile.

And just as an experiment let us see what happens if I make the line width for now so, we will we will keep this as a square hold on up to this there are no differences it is the same as in the earlier case except that it would be preferable if, we also define a not 101 but that as a different variable so let me define the value of some capital n as 101 let me replace all this by n right, see zeroes(N,1) and we are going to now say that.

(Refer Slide Time: 14:36)

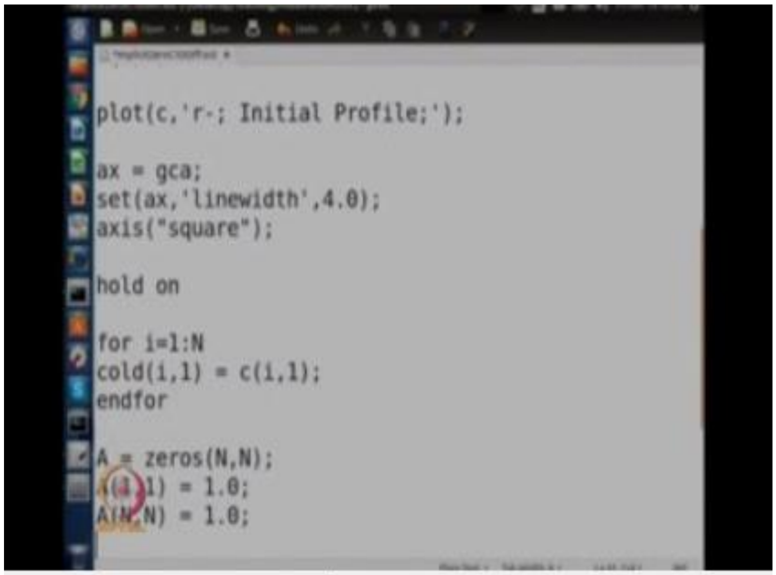


So we need to remember so when we use this implicit scheme we have to also define a see old right, i,1 = c(I,1); okay so this loop is extra this is something that we do not see in the other method but in the implicit scheme you need to keep track of which is in the old time which is in the new time because they are going to mix up when you do this matrix calculation, so it is better for you to define old and make that as the current one and then calculate a new one okay.

So the new calculated will become again old and then the next step will be taken and so on and so forth so it is important to have this defined so that is what we have done we have run a loop in which we have taken the sea and they also defined the old see at the first step the initial profile becomes the old after that, of course it is going to be updated as the calculation is going to proceed okay, so up to this set is the same so then we move on to the calculation so what does the calculation involved?

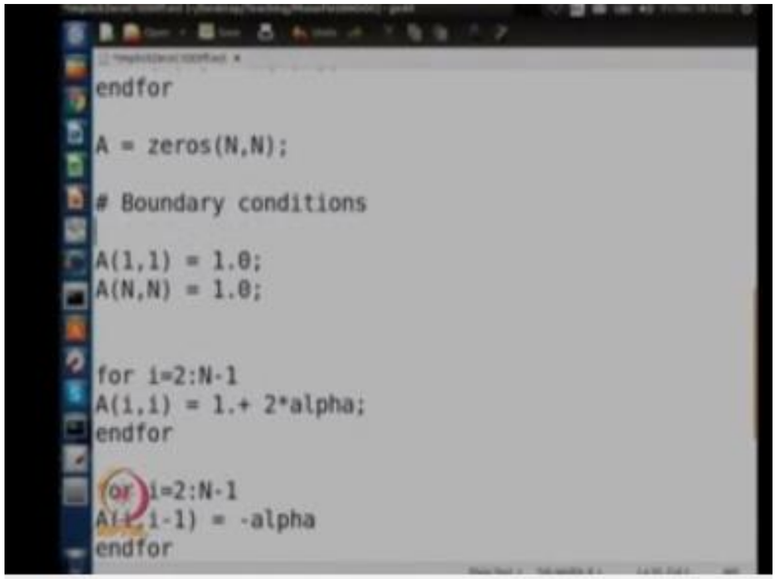
Now we have to define a matrix a is zeros(N,N) that is a size that is why it is better to define n because you do not have to keep writing 101 everywhere, so it will make a matrix which has zeros everywhere we know that a, so we need to run a loop so A(1,1) =1 this is the implementation of the divisional boundary condition at the left end A(101, 101) which is A(N,N) suppose right, so we can write it is as A(N,N) is also equal to 1.0 because you know nothing else is going to change.

(Refer Slide Time: 16:25)



In these two cases then the making of the matrix is very simple so then we run a loop for i i=1,2 and 2,2 because 1 is already taken 2 to N-1; A(i, i)=1,+2*alpha right, that is the diagonal term and far and then we want to write the lower diagonal at the upper diagonal, so for j =3 to2 N-1 for i =2 to N-1;A of so we need to write (I,i-1) right, so that is what – alpha and you have to say and for, similarly for i = we have 3 N and N for this case, we do the other way.

(Refer Slide Time: 18:07)



So A(i-1,i) = - alpha okay, so let us I mean it's a good idea to put a comment here so these are basically boundary conditions and this is diagonal terms, so this is basically lower this is up okay. So we have defined the matrix and as usual be run from k=1to 20 A= 1 to 500 then we are going to say c is equal to A, so let us try the other way this is what we did yesterday right, c a /a old is what we did yesterday, let's try this c is equal to inverse of a multiplying c old.

Then we need this other calculation for i = 1 to n, we need to say cold (i, 1) end for right, because we need to take the old values and we need to store them as the new okay. So that loop is done then there is N for which is the loop of a ends this J loop and then plot and the last end for is for ending the loop that is used for plotting and then of course we want to save this figure as something else, okay implicit 0 c dot eps okay, so I'm going to save this file okay and we go back and we run so, all i need to change is the change of the name.

So let us run and let us see what happens okay, okay the figure is cleaned so earlier figure is gone so clf has activated that means it will now do the calculation and the plot B so somewhere I have forgotten to put a semicolon okay, that is the reason why does this and you can see that what the figure as you can see this time the bounding box for the plot is a much thicker right toward then this time you use to force so it becomes much thicker and ,so the composition profile initial profile is here and then as time goes by it develops these profiles and it reaches a steady states.

As you can see the distance between these points also becomes smaller and smaller as you go towards this which is for the same reason as we discussed yesterday in the last lecture when you have a higher concentration gradients, you will have larger flux so things will happen faster but for the same time interval when the concentration gradients become smaller and smaller you know the concentration gradient here compared to here is very small so which means that the flux is also going to become smaller because of which the evolution with time is also going to slow down and which is what we are seeing.

So to summarize till now we have looked at two different cases the implicit and explicit and we have written codes for doing using implicit method or the explicit method in one dimensions for two types of boundary conditions both the boundary conditions are derris layer or one derris layered one moment because robin is not something that we will use it becomes very important if you are solving the heat transfer kind of problems but mass transfer diffusion, we will not worry about it.

The next thing that we want to do is to look at what is known as the periodic boundary condition so which is what I want to take up next and we will write a small whole for octave to solve for a case when we impose periodic boundary condition, so in the next part of this lecture I will discuss what is the periodic boundary condition and how we implement it and we will look at one sample problem to see what happens when you implement periodic boundary condition oka. Thank you.

NPTEL
Principal Investigator
IIT Bombay

Prof. R.K Shevgaonkar

Head CDEEP

Prof. V.M Gadre

Producer

Arun Kalwankar

Digital Video Cameraman
&Graphics Designer

Amin B Shaikh

Online Editor
&Digital Video Editor

Tushar Deshpande

Jr. Technical Assistant

Vijay Kedare

Teaching Assistants

Arijit Roy

G Kamalakshi

Sr. Web Designer

Bharati Sakpal

Research Assistant

Riya Surange

Sr. Web Designer

Bharati M. Sarang

Web Designer

Nisha Thakur

Project Attendant

Ravi Paswan

Vinayak Raut

NATIONAL PROGRAMME ON TECHNOLOGY ENHANCED LEARNING

(NPTEL)