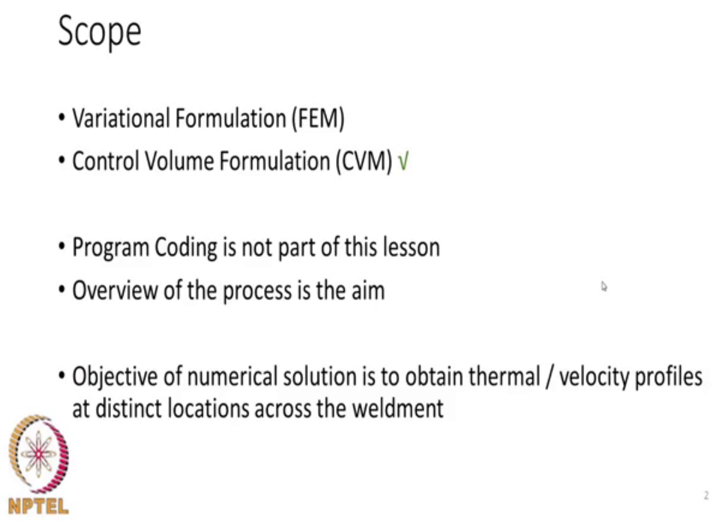
Analysis and Modeling of Welding
Prof. Gandham Phanikumar
Department of Metallurgy and Material Science
Indian Institute of Technology, Madras

Lecture - 18
Numerical solutions to thermal field and fluid flow in welding - Part 1

Welcome to the first part of the lesson on Numerical solutions to thermal field and fluid flow in welding. This lesson is part of the NPTEL MOOC on Analysis and Modelling of welding. My name is Gandham Phanikumar, I am from the Department of Metallurgy and Materials Engineering, IIT Madras.

(Refer Slide Time: 00:32)



Scope

- Variational Formulation (FEM)
- Control Volume Formulation (CVM) √

- Program Coding is not part of this lesson
- Overview of the process is the aim

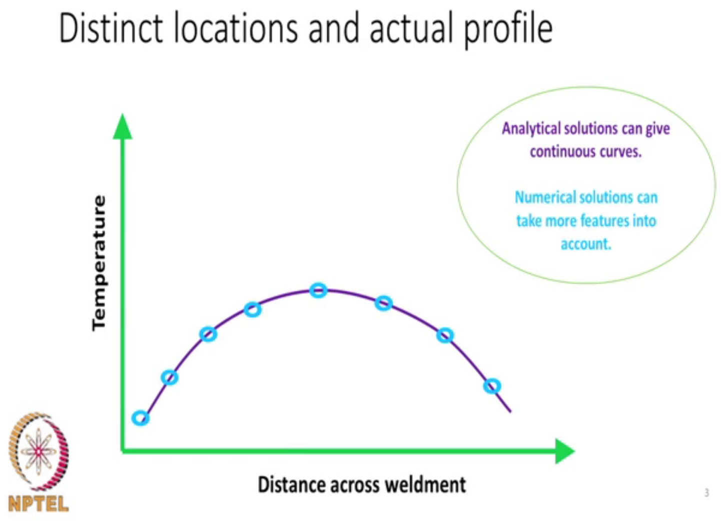
- Objective of numerical solution is to obtain thermal / velocity profiles at distinct locations across the weldment

The scope of this lesson will be a limited to a one of the two major classes of methods for numerical solution of the Navier-Stokes and generalized Fourier heat conduction equation that we have discussed and derived in earlier parts of this course. There are two major classes of numerical solutions; the first class is based on the variation formulation, and these are the methods that are used for FEM type of calculations. And many commercial software which use thermal and distortion calculations usually adopt this method.

And then there is a second class of numerical tools which are based on the control volume method, which can also be related to the finite difference methods, and these are very popular where the fluid flow and heat transfer are going to be considered. So, we are going to look up the solutions using the second method, namely control volume method mainly because that is a method I have used to derive the equations, and it is also a method where the flux balance is going to come out very naturally.

And we are going to go through the details in a depth that will give you an appreciation of how much attention to details is necessary, before you can simulate these processes using a computer. And the scope does not involve you to write a program out of this course mainly because that would take a lot of time and effort, and I would normally like you to consider that option once you have finished this course, and if you are interested to develop your own program. And the objective of numerical solutions is basically to obtain thermal and velocity profiles in the welding scenario. And we would like to have the results analyzed by knowing what are all the details that going to the simulation as you go ahead.

(Refer Slide Time: 02:27)



Distinct locations and actual profile

Analytical solutions can give continuous curves.

Numerical solutions can take more features into account.

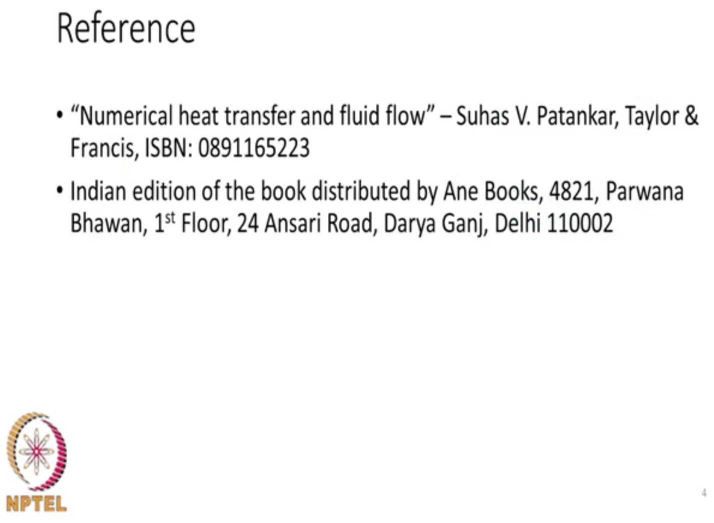
So, we have looked at analytical solutions earlier in this course. And we have seen that there are a number of methods that are available including for example, the popular rosenthal solutions. The analytical solutions are very valuable because they can give you answers to the thermal field at every location in the domain if you wish and which means that the solutions can be obtained as smooth curves. And you can see those plots for example; the violet line that is shown in this plot could have come from one such analytical solution.

However, the limitations are already discussed earlier analytical solutions are subject to a number of limits, for example, they do not take into account the variety of heat sources that are possible in welding, and the variety of heat removal processes that take place. And also fluid flow completely avoided in analytical solutions, because it is practically impossible to derive analytical solutions in welding taking the fluid flow into account.

Therefore, if want a more realistic solution for welding, then you must go to the numerical solution procedure; and usually when we pick the numerical solution, we would like to take them at discrete locations within the domain. The locations can be chosen a priori or adapted to the solution that is emerging in the domain, but we must know that we do not have the solution available at every single location in the domain, but at discrete locations and this is illustrated in the plot temperature versus distance across weldment.

You can see a curve over which there are some points that are given. You could thing that the curve may represent either an analytical solution or the exact solution that is actually prevalent. And the circles in blue are represent in the numerical solution that we would like to generate as part of our solution; and the difference between numerical analytical is that the analytical solutions can give you smooth curves while as numerical solutions give solutions for where the features more comprehensive.

(Refer Slide Time: 04:31)



Reference

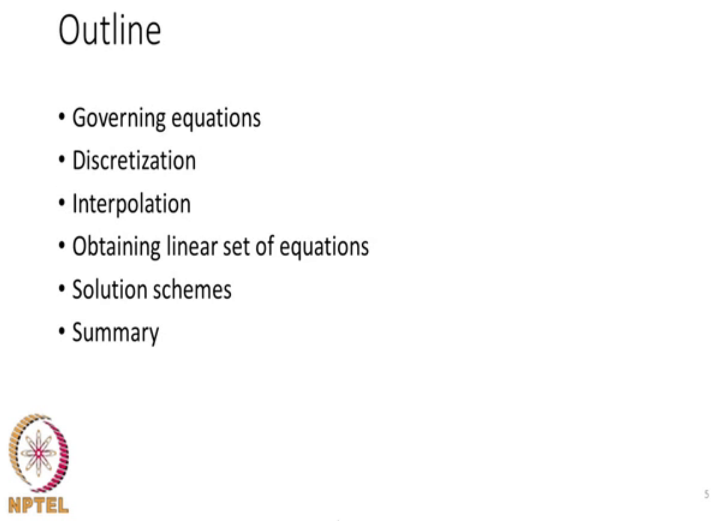
- "Numerical heat transfer and fluid flow" – Suhas V. Patankar, Taylor & Francis, ISBN: 0891165223
- Indian edition of the book distributed by Ane Books, 4821, Parwana Bhawan, 1st Floor, 24 Ansari Road, Darya Ganj, Delhi 110002

4

The ongoing discussion in this lesson and the following lesson to cover the numerical solutions will be referred by this book Suhas Patankar book on Numerical heat transfer and fluid flow.

This book is very important in the area of the control volume method to solve the fluid flow equations. And I would strongly recommend you to have a copy of this if you are planning to go further into this particular subject. And this book is also not very thick, so you should be able to go through that quite soon. And it is also going to give you hands on experience on how to write the program, because the expressions that are used are readily programmable in a language such as Fortran. And there is Indian addition also available which is not very expensive from Ane books.

(Refer Slide Time: 05:22)



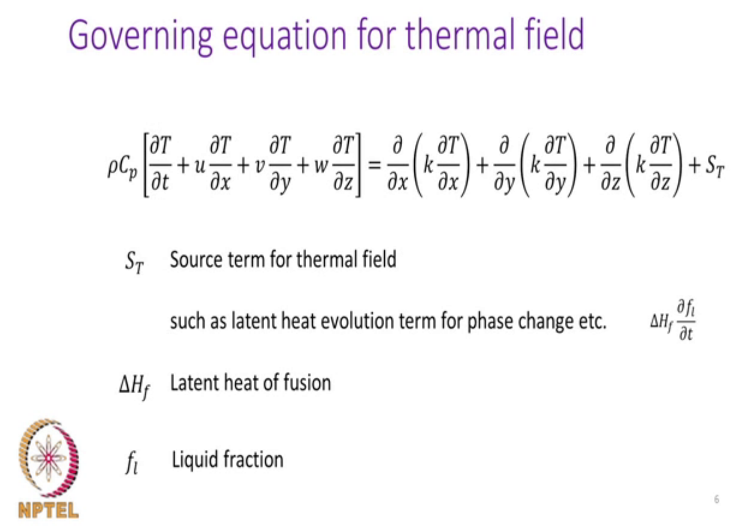
Outline

- Governing equations
- Discretization
- Interpolation
- Obtaining linear set of equations
- Solution schemes
- Summary

The outline of my lesson today is going to cover the following aspects. Governing equations we will just refer to the equations that we have derived earlier. And then convert them to a form that is generic for numerical solutions. And then we will see how to discretize the governing equation each of the terms will discretized, and we will see how we can use it to write the differential terms as properties of different locations.

And then we would see how to interpolate various parameters at intermediate locations and that is where a lot discussion will come when we come to the advection term in the next lesson. And then once we have interpolated and written the differential terms as discretized then we will able to obtain what are called a set of linear equations which could then be used to solve to obtain the solutions. So, the solution method also will be discussed, and finally, we will summarize our lesson in the end.

(Refer Slide Time: 06:21)



Governing equation for thermal field

$$\rho C_p\left[\frac{\partial T}{\partial t}+u\frac{\partial T}{\partial x}+v\frac{\partial T}{\partial y}+w\frac{\partial T}{\partial z}\right]=\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right)+\frac{\partial}{\partial z}\left(k\frac{\partial T}{\partial z}\right)+S_T$$

S_T Source term for thermal field

such as latent heat evolution term for phase change etc. $\Delta H_f\frac{\partial f_l}{\partial t}$

ΔH_f Latent heat of fusion

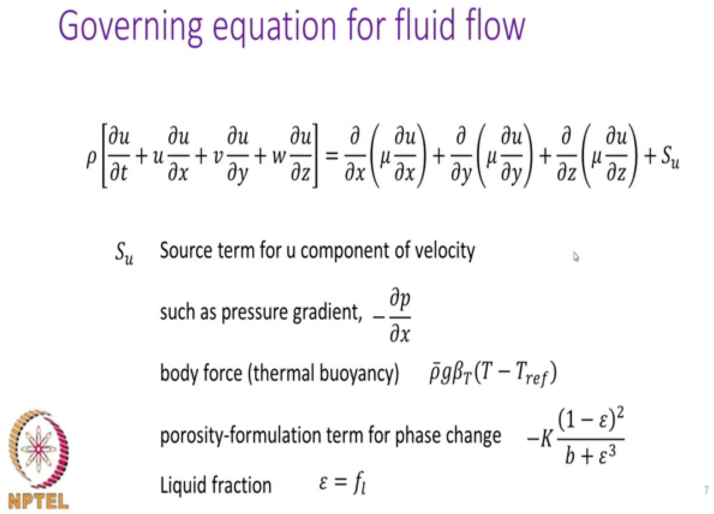
f_l Liquid fraction

6

The governing equations we have seen earlier the one I am writing here is for the thermal field, you can see that in this governing equation, we have got varies terms we have got already gone through this. So, we have already seen the governing equation that we have derived in the earlier lessons, and I am showing it to here for the thermal field. The first term is the transient term; the next three terms are the advective terms; and then on the right hand side, you have the first three terms referring to the diffusive term, and the last term is the source term.

And this source term essentially will take into account the latent heat, if the phase change is being considered as part of weld modelling, and the expression is for example, delta H f, where delta H f is the latent heat of fusion and dou f l by dou t, where f l is a liquid fraction. So, this equation is coming directly from the generalized Fourier heat conduction equation, which we are applying for welding.

(Refer Slide Time: 07:16)



Governing equation for fluid flow

$$\rho\left[\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}\right]=\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial u}{\partial z}\right)+S_u$$

S_u Source term for u component of velocity

such as pressure gradient, $-\frac{\partial p}{\partial x}$

body force (thermal buoyancy) $\bar{\rho} g \beta_T (T - T_{ref})$

porosity-formulation term for phase change $-K\frac{(1-\varepsilon)^2}{b+\varepsilon^3}$

Liquid fraction $\varepsilon = f_l$

And we also have derived by analogy the fluid flow equation the namely the Navier-Stokes equation applicable for fluid flow in the fusion zone of the weldment. And here also the first term is the transient term; the next three terms on the left hand side are the advective terms.

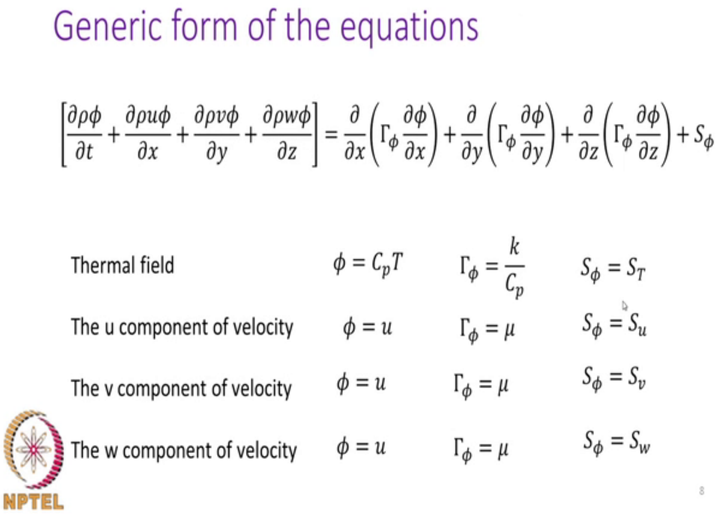
On the right hand side, we have the first three terms referring to the diffusive process of momentum diffusivity given by mu by rho; and then we have the last term, which is basically the source term. And this equation is written for the u component of the velocity which means that we will have two more such equations for the w component, and the v component. And a set of these three equations will then completely describe the fluid flow in the fusion zone.

And the source term is here going to have various terms depending upon the phenomena. We would normally have the pressure gradient term, where the gradient will be in the direction of the component of the velocity for which we are writing the equation. And then we will also have for example, a body force terms referring to for example, thermal buoyancy or solutal buoyancy. So, one example is given here for thermal buoyancy, rho bar the average density g beta T which is the expansion coefficient and then T minus T ref, where the T ref is the reference temperature normally it is chosen as the melting point.

So, we have the body force terms coming in as part of the source term - S u, and we also have for example, source terms coming to handle the change of the phase from liquid to solid. So, we have already discussed earlier that we want to write any equation that is valid from the entire domain, so that we can have single domain equations. And to handle that to ensure that the velocity would go to 0 in the solid, we wanted to use the porosity approach because it is very elegant and here is the term that refer to from that approach and is would be also coming as part of the source term.

And here the epsilon, I have written as it is from that formulation, it is same as the liquid fraction because that is the fraction through which the liquid can flow.

(Refer Slide Time: 09:20)



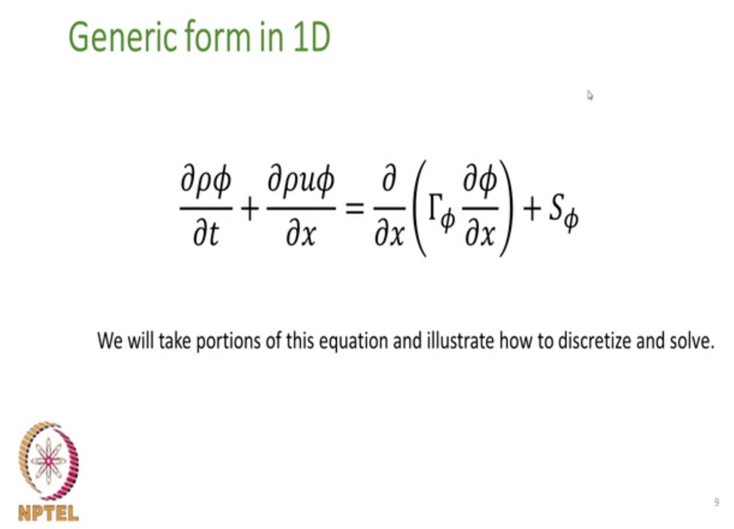
Generic form of the equations

$$\left[\frac{\partial \rho\phi}{\partial t}+\frac{\partial \rho u\phi}{\partial x}+\frac{\partial \rho v\phi}{\partial y}+\frac{\partial \rho w\phi}{\partial z}\right]=\frac{\partial}{\partial x}\left(\Gamma_\phi\frac{\partial \phi}{\partial x}\right)+\frac{\partial}{\partial y}\left(\Gamma_\phi\frac{\partial \phi}{\partial y}\right)+\frac{\partial}{\partial z}\left(\Gamma_\phi\frac{\partial \phi}{\partial z}\right)+S_\phi$$

Thermal field	$\phi = C_pT$	$\Gamma_\phi = \frac{k}{C_p}$	$S_\phi = S_T$
The u component of velocity	$\phi = u$	$\Gamma_\phi = \mu$	$S_\phi = S_u$
The v component of velocity	$\phi = u$	$\Gamma_\phi = \mu$	$S_\phi = S_v$
The w component of velocity	$\phi = u$	$\Gamma_\phi = \mu$	$S_\phi = S_w$

So, if you notice these two equations they are very similar, and therefore, we could write them as very generic form here. The generic form is written with phi as the variable, so that this can take different forms for different component to the velocity. So, you could see that the phi component term referring to the transient term, the second three are for the advective; and then on the right hand side, we have diffusive and the source terms. And for different fields thermal and the fluid flow, what are the forms that the gamma and S will be taken is given in this particular table.

(Refer Slide Time: 09:54)



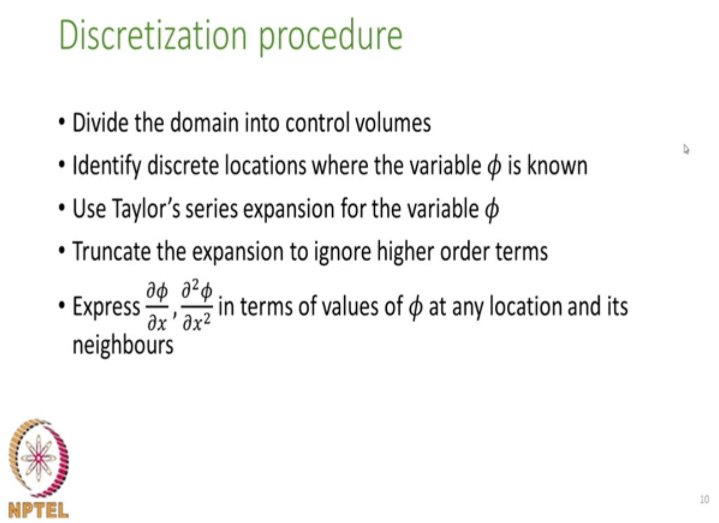
Generic form in 1D

$$\frac{\partial \rho \phi}{\partial t} + \frac{\partial \rho u \phi}{\partial x} = \frac{\partial}{\partial x}\left(\Gamma_{\phi} \frac{\partial \phi}{\partial x}\right) + S_{\phi}$$

We will take portions of this equation and illustrate how to discretize and solve.

And we can write the same equation in one dimension as follows. So, you could dropout the terms for the other two directions and you can see that this is a form that can be attempted to be solved numerically, and we will see it by term by term, so that we could get take first initially the diffusive term and then later on the advective term.

(Refer Slide Time: 10:16)



Discretization procedure

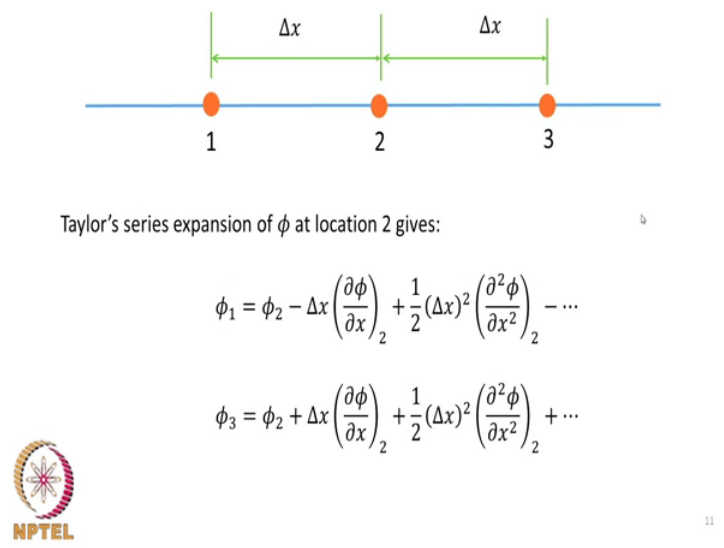
- Divide the domain into control volumes
- Identify discrete locations where the variable ϕ is known
- Use Taylor's series expansion for the variable ϕ
- Truncate the expansion to ignore higher order terms
- Express $\frac{\partial \phi}{\partial x}, \frac{\partial^2 \phi}{\partial x^2}$ in terms of values of ϕ at any location and its neighbours

And what we mean by discretization is a following process. We want to basically divide the domain into several control volumes. If you want to take a 1D domain, essentially we want to divide a line in two segments; and if we to take the 2D domain, we want to divide an area into squares or rectangular. And then if you are taking the three-dimensional domain, we want to divide them into cubes or cuboids. And each of these elements will then be analyzed for what is a flux that is going through their phases. And we want to also identify at what locations within each of this control volumes, any variable phi is known.

So, is it that the center of the control volume or is it at the control of the phases of the control volume is something that we need to decide, and we will see that both will be used for different parameters as we go along. And then whenever we want to express the value of the parameter phi, at any location other than the location where it is specified, then we need to interpolate; and the way we interpolate will be borrowed from the Taylor series expansion.

Essentially we will know the variable phi at a given location and then we will expand the phi as a smoothly varying function around that location and then use only the first order terms to see what would be the value at a neighboring location where we need interpolate. And then we will then use this expansion to derive a various slopes of this parameter phi, a first order slope, and the second order slope namely dou phi dou x and dou square phi dou x square. So, this is how we will be able to write the different terms of the 1D convective diffusive equation which we have seen earlier.

(Refer Slide Time: 11:57)



Taylor's series expansion of ϕ at location 2 gives:

$$\phi_1 = \phi_2 - \Delta x \left(\frac{\partial \phi}{\partial x}\right)_2 + \frac{1}{2}(\Delta x)^2 \left(\frac{\partial^2 \phi}{\partial x^2}\right)_2 - \cdots$$

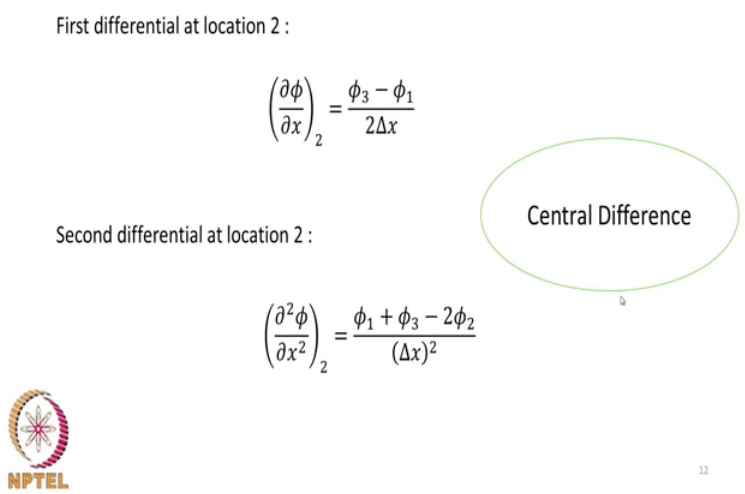
$$\phi_3 = \phi_2 + \Delta x \left(\frac{\partial \phi}{\partial x}\right)_2 + \frac{1}{2}(\Delta x)^2 \left(\frac{\partial^2 \phi}{\partial x^2}\right)_2 + \cdots$$

11

This is where we are showing you how we do the discretization. So, let us say that the variable phi is specified at the location 2. And we want to find out what would the value of the phi at location 1 or at location 3. So, what we can do is that if it is known at location 2, then in the vicinity location 2, we can expand the variable phi as if it is a smoothly varying function using the Taylors expansion, the first expression shows you how to do that.

Essentially we can see that the value of the variable at the location where it is specified that is given as phi 2 minus delta x dou phi by dou x evaluated at the location 2 plus half into delta x square dou square by dou x square evaluated the location 2. So, everything is evaluated at the location where the variable is specified. And you could do that to expand in the left hand side and the right hand side directions to show the distances as minus delta x and plus delta x. Now once you have these two equations, then you can add them and subtract them to get different terms.

(Refer Slide Time: 13:00)



First differential at location 2 :

$$\left(\frac{\partial \phi}{\partial x}\right)_2 = \frac{\phi_3 - \phi_1}{2\Delta x}$$

Central Difference

Second differential at location 2 :

$$\left(\frac{\partial^2 \phi}{\partial x^2}\right)_2 = \frac{\phi_1 + \phi_3 - 2\phi_2}{(\Delta x)^2}$$

And that is how we have written the slope, and the slope of this loop namely dou phi by dou x and dou square phi by dou x square evaluated at the location 2 as a function of the value of the parameter phi at the neighboring locations. And this is same as central difference method, and you can see that the slope is given by phi 3 minus phi 1 by 2 times delta x. If you were to write it only with the value at phi 2 and phi 1, then you would call it as forward difference; and if you were to write it in terms of phi 2 and phi 3, you would call it as backward difference.

So, what we have written here is the central difference method which is known to have better accuracy then the forward or backward difference methods. And the second differential is then written in terms of this slopes at either ends of the control volume phase, so you can write it as phi 1 plus phi 3 minus 2 phi 2 by delta x square.

(Refer Slide Time: 14:07)

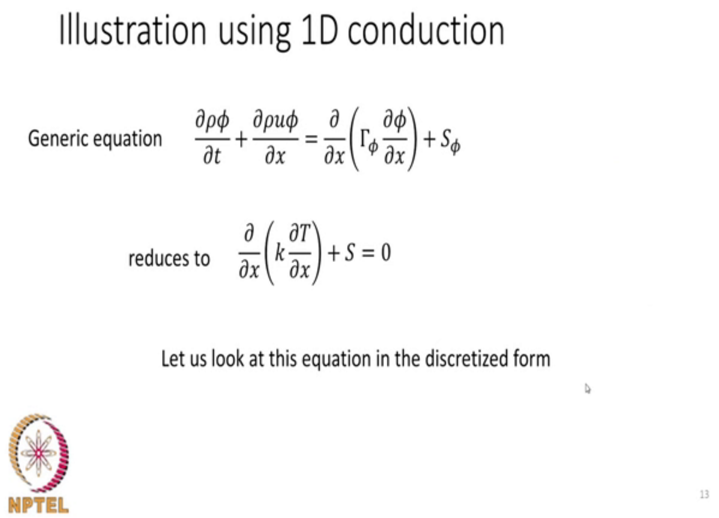


Illustration using 1D conduction

Generic equation $$\frac{\partial \rho\phi}{\partial t}+\frac{\partial \rho u\phi}{\partial x}=\frac{\partial}{\partial x}\left(\Gamma_{\phi}\frac{\partial \phi}{\partial x}\right)+S_{\phi}$$

reduces to $$\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right)+S=0$$

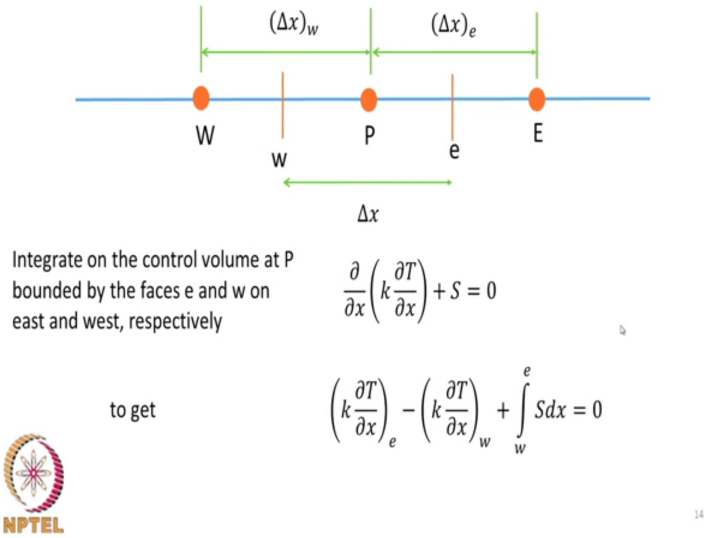
Let us look at this equation in the discretized form

13

So, now that we have the differentials written in terms of the value of the parameter phi at discrete locations 1 to 2 etcetera which are then given in the domain then we can go further to see how the differential equation is going to look like. So, the generic equation is written here and then we want then apply it for a specific case.

To illustrate, we will choose the case of only heat conduction, so that we can just take terms on the right hand side first. We are taking the diffusive term and the source term, and then we are basically trying to solve this in a numerical manner without considering the advective and the transient term. So, you can say that what we attempting now is steady state 1D heat conduction with source term using a numerical method.

(Refer Slide Time: 14:41)



Integrate on the control volume at P bounded by the faces e and w on east and west, respectively

$$\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right)+S=0$$

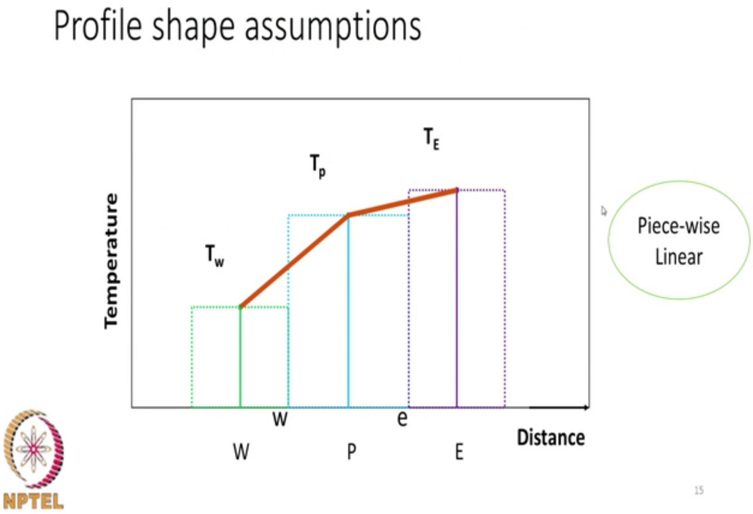
to get

$$\left(k\frac{\partial T}{\partial x}\right)_e-\left(k\frac{\partial T}{\partial x}\right)_w+\int_w^e Sdx=0$$

So, let us see how the method would evolve. So, we are basically looking at this equation, and we will see that P is used always to indicate the location at which we want evaluate. And the control volume is having the faces on the east side and the west side detonated as e and w. The vertical lines are showing you the control volume faces. And this is a control volume over which we are going to essentially integrate this particular differential equation. And when you integrate you already have a differential, so you could see that integration of the first term will give you k dou T by dou x at east face minus west face that would be the integral of the first term.

The second term will be integrated in this manner. And the source term could be constant within the control volume or it may vary; in either case we would be taking an average value within the domain, so that the second term here for the source term integration can be taken as an average value multiplied by the delta x that is the width of the control volume.

(Refer Slide Time: 15:40)



Profile shape assumptions

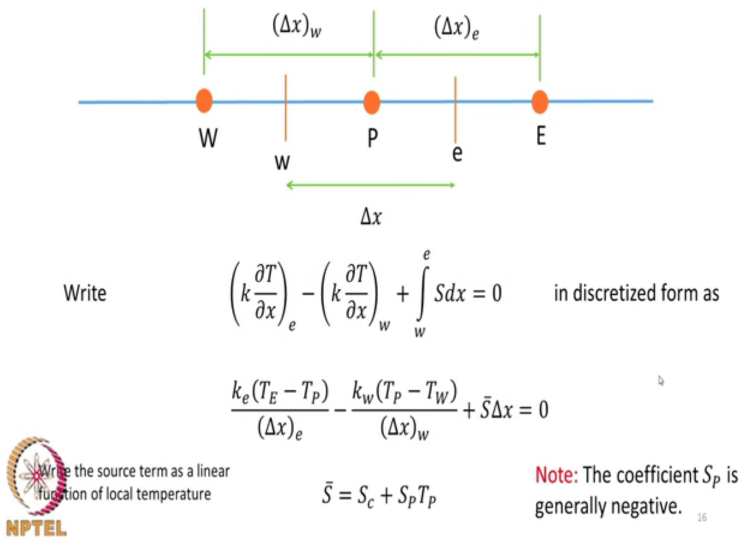
Piece-wise Linear

15

So, we would do that now; and while we do that, we also need to implicitly assume how would the parameter phi or in this case temperature vary across the control volume and beyond. So, what we are doing basically is a piece-wise linear variation which is reasonable; and this is different from another assumption that is possible which is basically to choose that the temperature is constant in the entire control volume. We are not choosing that assumption because that would lead to a sudden jump in the temperature across the base of the control volume; we are not doing that.

What we are doing is that we are saying that the temperature is assume to be specified at the center of the control volume; and across to control volumes, it is varying in a piece-wise linear form, so that then we want a temperature at intermediate locations. We can do interpolation linear interpolation and we can get the temperatures at those locations.

(Refer Slide Time: 16:35)



Write $\left(k\frac{\partial T}{\partial x}\right)_e - \left(k\frac{\partial T}{\partial x}\right)_w + \int_w^e S dx = 0$ in discretized form as

$$\frac{k_e(T_E - T_P)}{(\Delta x)_e} - \frac{k_w(T_P - T_W)}{(\Delta x)_w} + \bar{S}\Delta x = 0$$

Write the source term as a linear function of local temperature

$$\bar{S} = S_c + S_P T_P$$

Note: The coefficient S_P is generally negative.

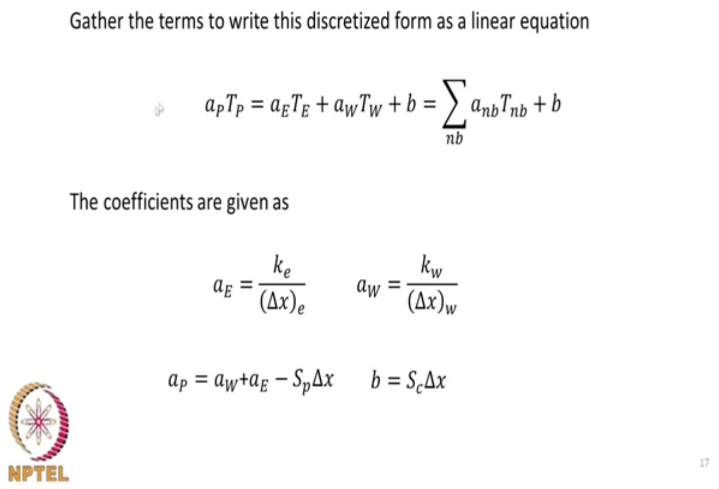
And if we do that then what we can then assume is that dou T dou x can then be given by the Taylor's series expansion that we had discussed earlier, so that a linear approximation is possible and you could then look at it like this. The slope at e is given by capital e minus capital p that is temperature at this location minus temperature at this location divided by the distance here which is delta x e. So, you can see that the first term is discretized to be just difference of the temperatures divided by the distance.

Similarly, the second term also has been discretized. And then the integral of the source term is then taken as an average value of the source term across entire control volume multiplied by the width of the control volume. Now once you have this then you can then proceed to gather the terms.

And one small clarification here, in case this source term is a function of temperature, then necessarily we would like to linearise it in this form S c plus S p times T p. The S p does not indicate evaluation of the source term at the location p; it only indicates the coefficient of the source term with respect to the temperature. So, this is basically what is called as linearised source term which means that if your source term is going to be non-linear function of temperature, we need to make it linear.

We have discussed this briefly in one of the lessons earlier and here we are seeing the reason why we need to get. And usually you have the S p having a negative value, so that the temperature evaluation will be stable and we would see that it is helping in one of the four rules that we will be talking about for the coefficients of the discretized equation that we are writing.

(Refer Slide Time: 18:15)



Gather the terms to write this discretized form as a linear equation

$$a_P T_P = a_E T_E + a_W T_W + b = \sum_{nb} a_{nb} T_{nb} + b$$

The coefficients are given as

$$a_E = \frac{k_e}{(\Delta x)_e} \qquad a_W = \frac{k_w}{(\Delta x)_w}$$

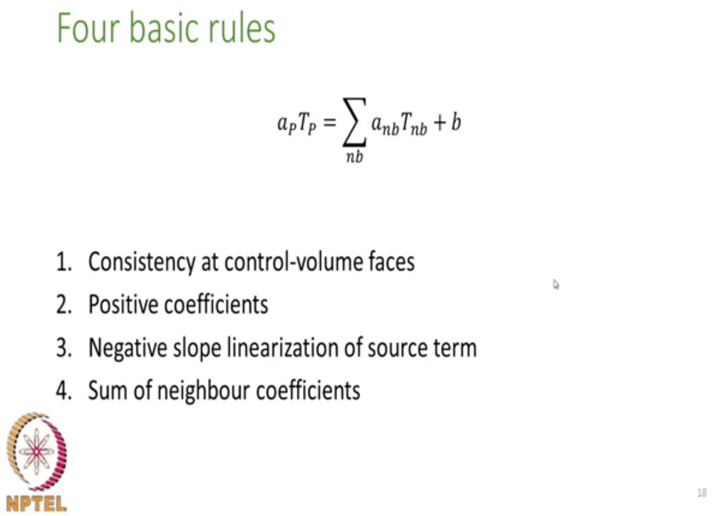
$$a_P = a_W + a_E - S_p \Delta x \qquad b = S_c \Delta x$$

And when we gather the terms this is how it would appear, you would write it as on the left hand side the temperature at location p, T p, and it is depended upon a weighted average of then temperature at the neighboring locations. And then there will be constant term that will be added which takes into account the source term. And you can see that it is a essentially coming as a summation of neighboring location temperatures weighted by a coefficient. And what are those coefficients; it has been seen from the previous slide that it comes as basically a ratio of the thermal conductivity and the distances.

And this way, we can now see that at any energy location we can evaluate what would be the temperature as a function of the neighboring locations. The coefficients are expanded here, you can that the coefficient of T e and T w are very simply given by the thermal conductivities and the distances on either ends of the locations, where we are evaluating.

Whereas the coefficient a p is given as summation of the all the coefficients minus the source term, which means that in the case where you do not any source term the coefficient a p will just be the summation of the coefficients a e and a w which is again a very important role. It will help us validate the discretization process for more complex equations that will come across later on.

(Refer Slide Time: 19:38)



Four basic rules

$$a_P T_P = \sum_{nb} a_{nb} T_{nb} + b$$

1. Consistency at control-volume faces
2. Positive coefficients
3. Negative slope linearization of source term
4. Sum of neighbour coefficients

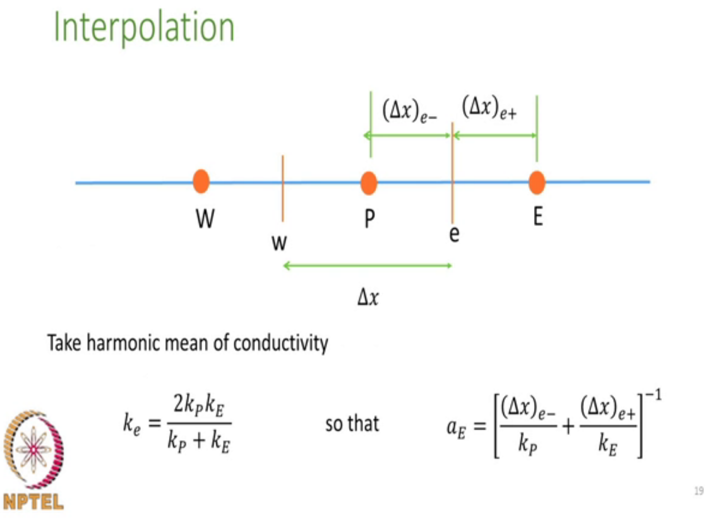
And there are four rules that we would be addressing when we write this equations. We must understand that control volume method is based upon the flux balance, which means at the face of the control volume the flux that is arriving should be the same as what is leaving, which means that we must definitely use the same mathematical expression to write the flux that is coming in which is balanced by the flux that is going out. And the piece wise linear relationship we have seen is going help in that.

In case, we chose for example, piece wise parabolic variation of temperature then it will not work out to be the same and you may actually have some numerical errors that will be coming up. So, it is very important to ensure that mathematically when we are write flux at the control-volume faces the expression is looking identical for both faces both sides of the face for two adjoining control volumes.

And all the coefficients must be positive you can just verify that from the expression for the coefficient you can see that it is given by thermal conductivity, which will be positive divide by distance which is also going to be positive. So, which means that all coefficients must be positive this is also going to help in evaluating, what would be the problem in case the discretization by using various expressions is going wrong for a more complicated situation. And we must ensure that all coefficients should be positive.

And then the negative slope of the linearization is also to be ensured and if that is ensured then you can ensure also that the sum of the neighboring coefficients will be equal to the coefficient of that T p in the absence of the source term. And once we follow these rules, then numerical scheme is guarantee to be stable and this is also a check to see if there is a problem with this scheme or with the solution process whenever we get absorbed results out of the numerical computation. So, we must first ensure that these rules are followed and later then see where could the problem be in case the results are not coming out to be nice.

(Refer Slide Time: 21:36)



Interpolation

Take harmonic mean of conductivity

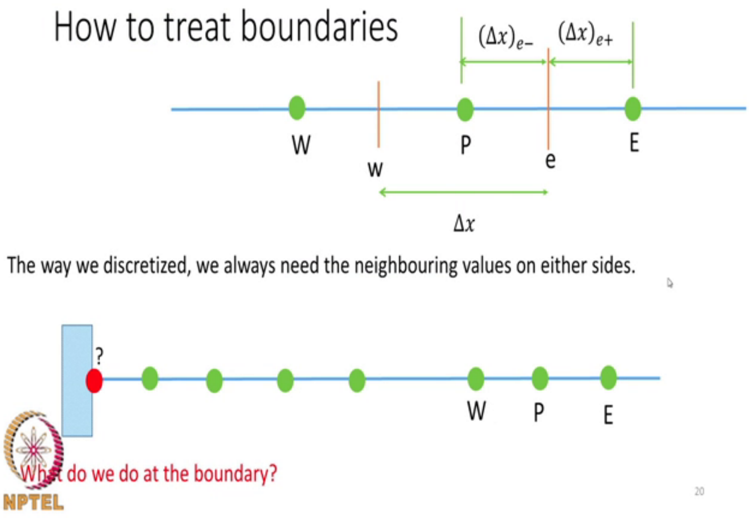
$$k_e = \frac{2k_P k_E}{k_P + k_E} \qquad \text{so that} \qquad a_E = \left[\frac{(\Delta x)_{e-}}{k_P} + \frac{(\Delta x)_{e+}}{k_E}\right]^{-1}$$

The interpolation scheme here is expanded here in a bit more and it is not as if we can take every parameter in the domain, and then make a linear interpolation. For example, take the thermal conductivity. Let us say the thermal conductivity we have the choice of using interpolations using either linear or harmonic. And here the argument is as follows let us say that the control volume to the left of P is having a thermal conductivity which is very poor compare to the control-volume in the right hand side. Then what would happen is that we should not have much of heat flux coming from the left to the location P.

However, if you were to do a an averaging of thermal conductivity, then the thermal conductivity at the location w here would be average between the two which means that if the thermal conductivity on the left hand side is 0, then you would still have a finite thermal conductivity at face w, which means that would be some heat flux that is going from the left to the right. It should not be possible in case the thermal conductivity at w is 0 and this kind of a situation can be avoided if you use actually harmonic mean. And that is where we are actually showing you here the thermal conductivity can be taken harmonic mean which means that the way you interpolate the thermal conductivity is to take this kind of an expression.

So, one must pay attention to evaluate in the variables at different locations in the domain, before we directly use the interpolation which is linear which is default for most of the parameters.

(Refer Slide Time: 23:12)



How to treat boundaries

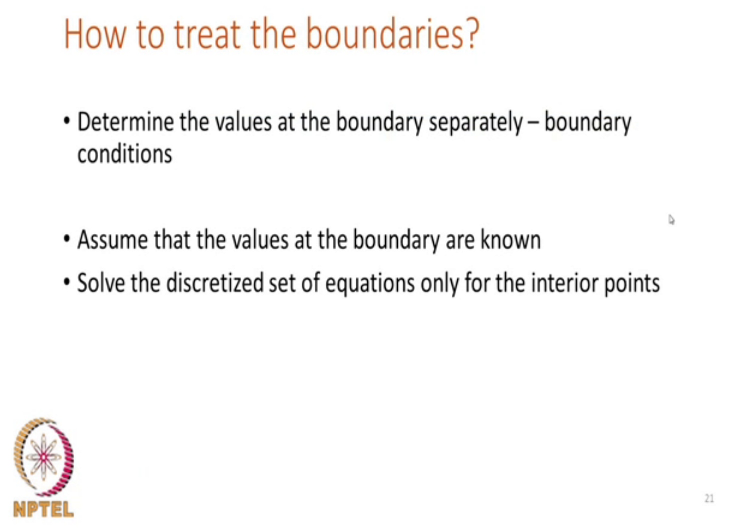
The way we discretized, we always need the neighbouring values on either sides.

What do we do at the boundary?

And this can be done for any location; however, when we come to the boundary we see that we need to treat the boundary little bit specially. The reason being as follows for any location we saw that the linear equation, we wrote out of the discretized form is going to contain the neighboring values, for example, for P; we saw that the expression will have the values involving W and E.

However, when you come to the boundary then we see that for the note that is exactly on the boundary you have a neighbor on the right hand side, but there is no neighbor on the left hand side, which means that we will not be able to use the same expression, which means that we need to treat it separately.

(Refer Slide Time: 23:49)



How to treat the boundaries?

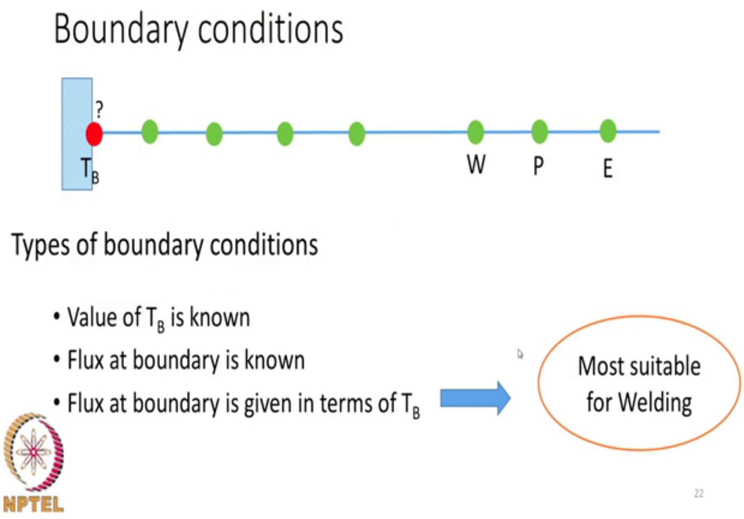
- Determine the values at the boundary separately – boundary conditions
- Assume that the values at the boundary are known
- Solve the discretized set of equations only for the interior points

And the way we treat it is like this. There are three ways of doing that. Essentially, we must have the value of the boundary known. The numerical solutions require that the value of any parameter phi in this case temperature is known at the boundary. And very often the boundary temperature may be specified, for example, if you have a large plate that is being welded and that domain is also a small portion of plate then the boundary temperature may be given as the ambient temperature itself.

In such situations, there is no problem; the boundary condition is well defined as a value boundary condition. However if that is not the situation and if you have for example, heat loss that is specified at the boundary, and if it is a constant heat loss, which is the second condition or for example, if you have heat loss that is given as a function of the boundary temperature then we need to do some more treatment. And we will see that in a movement.

And essentially we have to now agree that we will not solve the linearized set of equations on the boundary, we will use only the interior points for the solution and then we are going to use the boundary values separately. So, in other words, when you write a program you will have two routines; one routine that will be solving for the values of the variable T at all the interior points, which will be basically solution of the linear set of equations, and then you would have another routine where you will be evaluating what would be the boundary temperatures and that would be discussed now.

(Refer Slide Time: 25:17)



Boundary conditions

Types of boundary conditions

- Value of T_B is known
- Flux at boundary is known
- Flux at boundary is given in terms of T_B

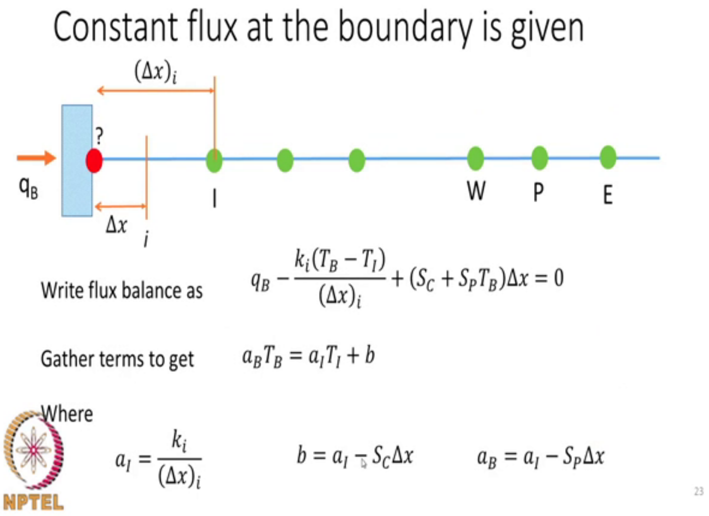
Most suitable for Welding

22

This is how we can do. There are three ways of doing the boundary conditions; one way is the boundary temperature itself is specified, and there is very simple and it is also applicable for welding where the domain is much smaller than the actual welding plate. The flux at boundary is known, and this is possibly in a situation where you have a constant heat plus that is being either given or removed from the boundary. But most popular condition in welding would be the third one where the flux at the boundary is given in terms of the boundary temperature. And this is valid for example, convective heat loss on the side wall and on the top. It is also valid for radiative heat loss on the surface and in all the situations the boundary temperature is playing a role.

And at the end of the analysis, we need to have T b known, so that for solution of the linearized set of equations for the location 2 onwards, you have the neighbors known. And the way we handle the second and third condition is as follows.

(Refer Slide Time: 26:12)



Constant flux at the boundary is given

Write flux balance as $q_B - \dfrac{k_i(T_B - T_I)}{(\Delta x)_i} + (S_C + S_P T_B)\Delta x = 0$

Gather terms to get $a_B T_B = a_I T_I + b$

Where

$a_I = \dfrac{k_i}{(\Delta x)_i}$ $\quad b = a_I - S_C \Delta x$ $\quad a_B = a_I - S_P \Delta x$

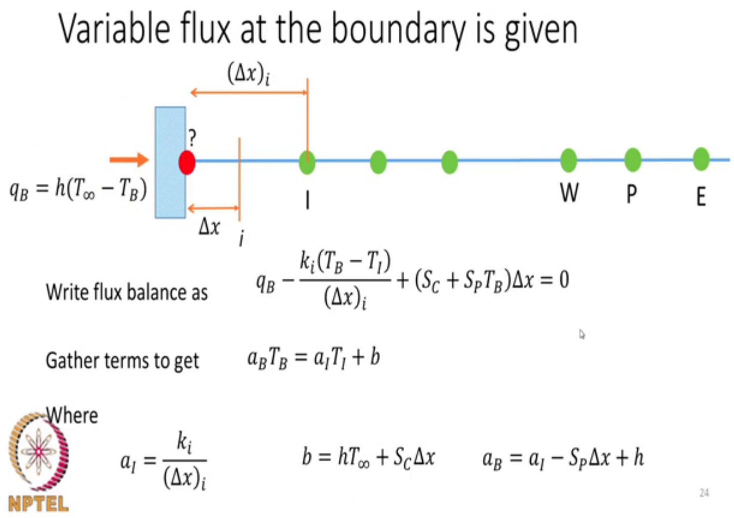
NPTEL

23

When the constant heat flux is given, what we do is we balance the heat flux at the boundary and obtain what would be the value of the temperature at boundary namely T b. So, the flux balance is given here as follows. The flux that is coming is q B and what the flux that is leaving is given by the Fourier heat condition first law and that is given here. And the source term is evaluated at the boundary as follows.

Then, once you gather these terms from the discretized from of a flux balance then you would see that again it is given as a B T B is equal to a i T i plus b; i is the interior point at which the temperature will be known. So, which means that you can use this equation to solve for T b, and all coefficient are also available. So, which means that we have a way by which we can find out the value at T B; and once T B is known, then our solution process can start.

(Refer Slide Time: 27:05)



Variable flux at the boundary is given

Write flux balance as $q_B - \frac{k_i(T_B - T_I)}{(\Delta x)_i} + (S_C + S_P T_B)\Delta x = 0$

Gather terms to get $a_B T_B = a_I T_I + b$

Where

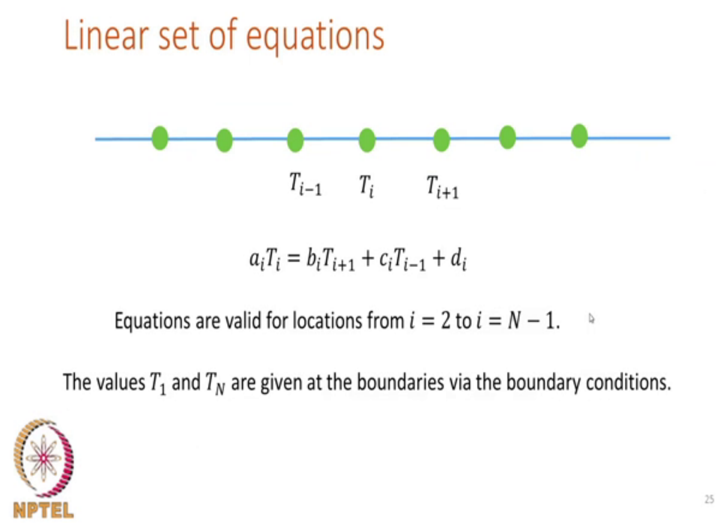
$a_I = \frac{k_i}{(\Delta x)_i}$ $b = hT_\infty + S_C\Delta x$ $a_B = a_I - S_P\Delta x + h$

24

In situations where the flux at the boundary is given as a variable with respect to the boundary temperature, for example, convective heat loss or heat gain then you would have the expression little bit more involved, and you could write the flux balance as follows you could write it in this manner. And the moment, you then substitute what is the expression for q b into this flux balance, you would see that the coefficient have changed their form. So, you could compare with the previous slide look at the value for B, for example, you would see that the B form has changed and you are seeing that the for field temperature T infinity is also coming into the solution of T B.

And you can use this expression to evaluate what the T B. And once we use these methods essentially whether it is constant heat flux or variable heat flux or constant temperature, we are finally achieving what would be the boundary temperature calculated separately; once that is known then we can go and try to solve the equations for the interior points.

(Refer Slide Time: 28:07)



Linear set of equations

$$a_iT_i = b_iT_{i+1} + c_iT_{i-1} + d_i$$

Equations are valid for locations from $i = 2$ to $i = N - 1$.

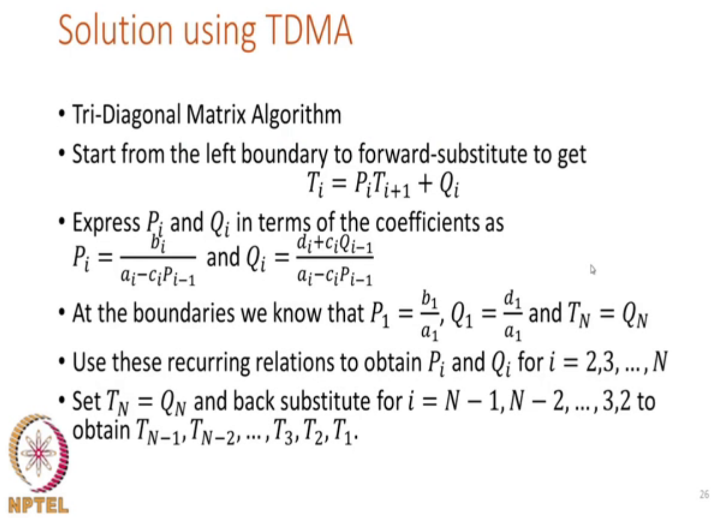
The values T_1 and T_N are given at the boundaries via the boundary conditions.

25

And that is then written as a set of linear equations. You see that the linear equations are written always as weighted averages of the neighboring temperature; in this case, for example, a i, b i and c i are the weights for the averaging and then a source term that is given in the end as d i.

So, all the equations are going appear in the same form; and these are going to solve from location 1 to N, where N is number of control volumes that we have divided the domain into what we are going from 2 to N minus 1, because i is equal to 1 and i is equal to N refer to the boundary. And these are then obtained from the boundary condition separately.

(Refer Slide Time: 28:46)



Solution using TDMA

- Tri-Diagonal Matrix Algorithm
- Start from the left boundary to forward-substitute to get

$$T_i = P_i T_{i+1} + Q_i$$

- Express P_i and Q_i in terms of the coefficients as

$$P_i = \frac{b_i}{a_i - c_i P_{i-1}} \text{ and } Q_i = \frac{d_i + c_i Q_{i-1}}{a_i - c_i P_{i-1}}$$

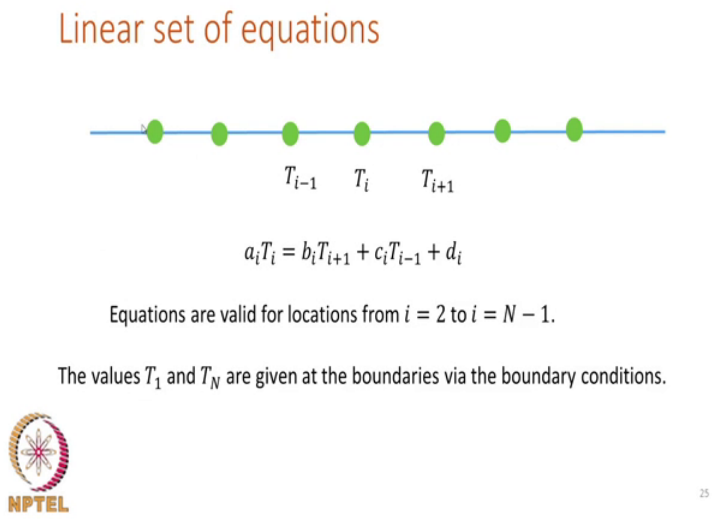
- At the boundaries we know that $P_1 = \frac{b_1}{a_1}$, $Q_1 = \frac{d_1}{a_1}$ and $T_N = Q_N$
- Use these recurring relations to obtain P_i and Q_i for $i = 2,3, \dots, N$
- Set $T_N = Q_N$ and back substitute for $i = N-1, N-2, \dots, 3,2$ to obtain $T_{N-1}, T_{N-2}, \dots, T_3, T_2, T_1$.

NPTEL

26

And how are we going to solve this equations, if you were write the these equations as matrix then you would see that you have a matrix in which only three diagonal rho are filled and therefore, you could use for example, a tri-diagonal matrix algorithm. This algorithm essentially is in two stages, you have what is called the forward substitution, and then backward substitution or backward evaluation.

(Refer Slide Time: 29:15)



Linear set of equations

$$a_i T_i = b_i T_{i+1} + c_i T_{i-1} + d_i$$

Equations are valid for locations from $i = 2$ to $i = N - 1$.

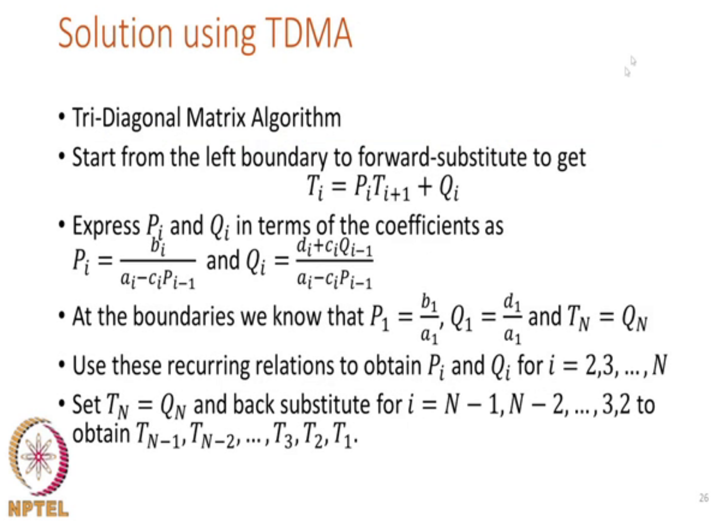
The values T_1 and T_N are given at the boundaries via the boundary conditions.

NPTEL

25

So, what we are going to do is as follows. Essentially, we are going to start from one end of the domain and because on left hand side we have the known value then you can use the equation to find out what is on the right hand side. And then you can then proceed to go further by one step, and again you have got the left hand side value know we can find out what is a right hand side, so that is the way we are going to do.

(Refer Slide Time: 29:33)



Solution using TDMA

- Tri-Diagonal Matrix Algorithm
- Start from the left boundary to forward-substitute to get
$$T_i = P_i T_{i+1} + Q_i$$
- Express P_i and Q_i in terms of the coefficients as
$P_i = \frac{b_i}{a_i - c_i P_{i-1}}$ and $Q_i = \frac{d_i + c_i Q_{i-1}}{a_i - c_i P_{i-1}}$
- At the boundaries we know that $P_1 = \frac{b_1}{a_1}$, $Q_1 = \frac{d_1}{a_1}$ and $T_N = Q_N$
- Use these recurring relations to obtain P_i and Q_i for $i = 2,3,\dots,N$
- Set $T_N = Q_N$ and back substitute for $i = N-1, N-2, \dots, 3,2$ to obtain $T_{N-1}, T_{N-2}, \dots, T_3, T_2, T_1$.

NPTEL

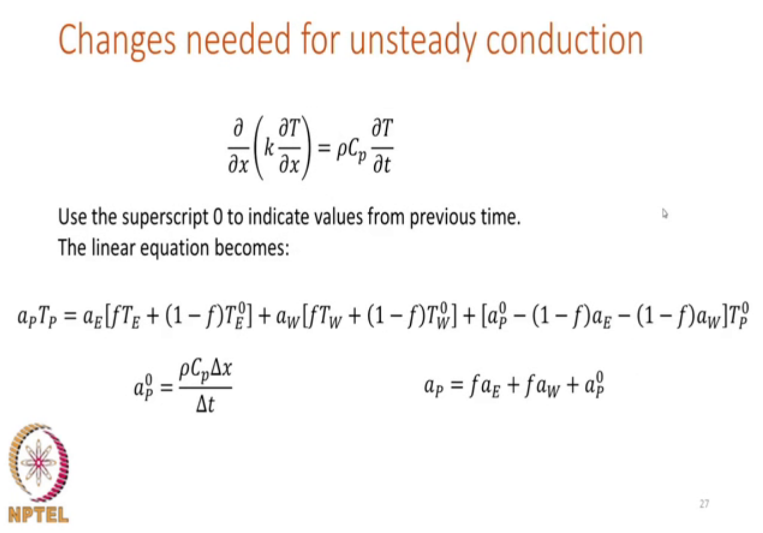
26

We are going to write this equation where the P i and Q i are going to be evaluated. And then once we go through the entire set of equations then you can start evaluating from the boundary to get what be the temperature. And therefore, by looking at the sequence here, you can see that from this linear set of equations by substituting, you can get the values from N minus 1 onwards backwards, and you can then find out the values of T N. Directly we obtain the values.

And you do not need to for example, invert the matrix to get the solution; inversion of the matrix is possible, in case you have 1D problem, you can definitely do that, but then you have a partially filled matrix then it is not efficient to invert the matrix because it would be computationally intensive and tri-diagonal matrix algorithm memory efficient, and it can give you the answers directly without having to have matrix inversion.

So, this is one algorithm that is freely available also as a source code if you want to barrow from the internet. And you could also use these equations to directly program it if you are interested.

(Refer Slide Time: 30:37)



Changes needed for unsteady conduction

$$\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) = \rho C_p \frac{\partial T}{\partial t}$$

Use the superscript 0 to indicate values from previous time.
The linear equation becomes:

$$a_P T_P = a_E[fT_E + (1-f)T_E^0] + a_W[fT_W + (1-f)T_W^0] + [a_P^0 - (1-f)a_E - (1-f)a_W]T_P^0$$

$$a_P^0 = \frac{\rho C_p \Delta x}{\Delta t} \qquad a_P = f a_E + f a_W + a_P^0$$

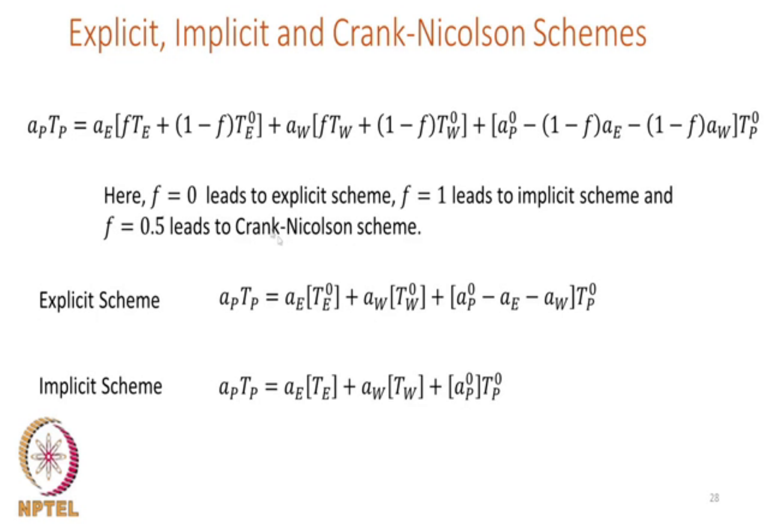
And once we have this then we can see what changes will be required in the coefficients, in case we are interested in unsteady conduction, which means that we are now adding one more term. And instead of the source term we are now trying to add the unsteady conduction.

The difference between unsteady conditions versus conduction at steady state it is as follows, we have basically a temporal variation which means that you now have that temperatures stored at different time steps. And at time T is equal to 0, we essentially have the initial condition that are being stored.

And the previous time step values are all given with 0 as the superscript, so that we can identify them. And essentially, you have the same kind of a discretization that is going to be applicable and when you do that and gather terms this is how the equation would look like.

And you have the variable f that is being used here, with the particular intention. This f essentially will tell you whether you are going to use for any temperature at the neighboring location, whether we are going to use it from the previous time step or from the current time step, so f is going to determine that and we are going to then talk about that in a moment.

(Refer Slide Time: 31:54)



Explicit, Implicit and Crank-Nicolson Schemes

$$a_PT_P = a_E[fT_E + (1-f)T_E^0] + a_W[fT_W + (1-f)T_W^0] + [a_P^0 - (1-f)a_E - (1-f)a_W]T_P^0$$

Here, $f = 0$ leads to explicit scheme, $f = 1$ leads to implicit scheme and $f = 0.5$ leads to Crank-Nicolson scheme.

Explicit Scheme $\quad a_PT_P = a_E[T_E^0] + a_W[T_W^0] + [a_P^0 - a_E - a_W]T_P^0$

Implicit Scheme $\quad a_PT_P = a_E[T_E] + a_W[T_W] + [a_P^0]T_P^0$

NPTEL

28

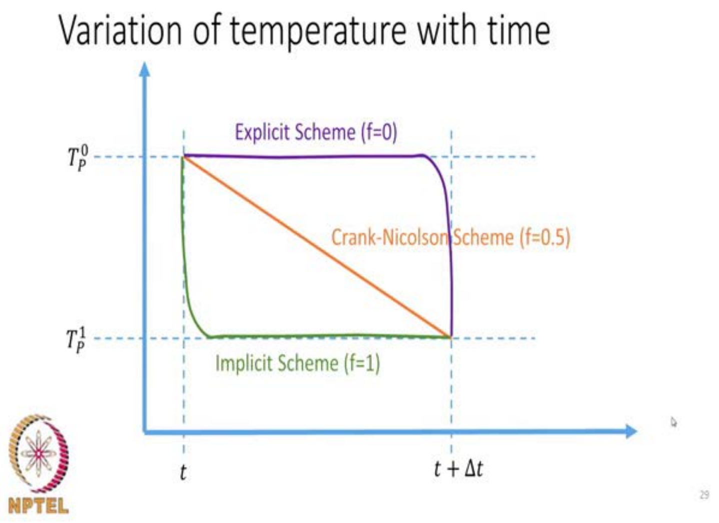
The coefficients are then evaluated in the same manner as described earlier. This f for example, if were to use f is equal to 0, and you can look at the condition here, if f where to be zero then the coefficient of a E is not the current time step temperature, it is a previous tempted temperature that, which means that we are going to use known values of neighbor temperatures from the previous time steps to evaluate the current location temperature at the current time step. And this would be called as an explicit scheme.

The reason being that in the scheme, you are basically going to evaluate temperature where everything on the right hand side is known. And if f is equal to 1, you would see that the previous time step temperatures of the neighboring locations are not used, you have only the current time step temperature that are being used for the neighbors, which means that it is called as an implicit scheme, which means that all the variable on the right hand side of the equation are also at the same time step and are unknown. It requires that then we have to iterate this equation to find out the values.

So, f is equal to 0 would give you the explicit scheme; f is equal to 1 will give you the implicit scheme. And in this expression, if you were to use a f is equal to 0.5 then you go into a special scheme called as Crank-Nicolson scheme where we have a 50-50 as usage of the previous and the current time step temperatures. And this can lead to some problems in some schemes, but generally it is a good mixer between the explicit and implicit schemes and this equation can then be a simplified for explicit and implicit as follows.

And you see that in the explicit scheme everything on the right hand side is available from the previous time step temperatures, and therefore, you can directly obtain what will the temperature at P in the current time step. In the implicit scheme, everything on the right hand side is actually also at the current time step and therefore, you have to iterate.

(Refer Slide Time: 33:45)

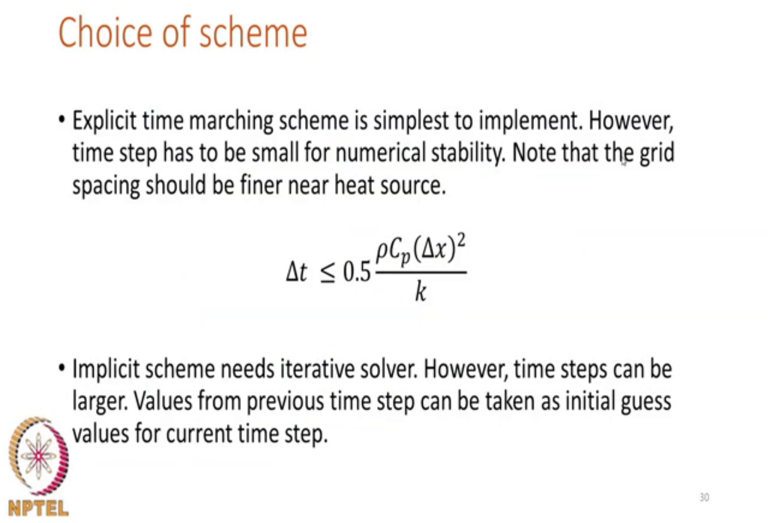


Variation of temperature with time

And how do these three schemes compare with each other. So, here we are plotting what would be temperature at location P, and you can see from time T to T plus delta T how the variation would be considered. Explicit scheme would mean that until we reach the current time step, we are assuming that the variable is changing taking the same value and Crank-Nicolson scheme is actually something in between. And you would see that explicit scheme is when f is equal to 0, implicit scheme when f is equal to 1 and Crank-Nicolson scheme is basically showing you, the variation across the two discrete time steps very smoothly going.

And these three schemes have to be discussed before we adopt one of them for our solution, and generally one would use implicit scheme in this problems because you do not have the time step problem whenever you want to do a long time simulation.

(Refer Slide Time: 34:39)



Choice of scheme

- Explicit time marching scheme is simplest to implement. However, time step has to be small for numerical stability. Note that the grid spacing should be finer near heat source.

$$\Delta t \le 0.5\frac{\rho C_p(\Delta x)^2}{k}$$

- Implicit scheme needs iterative solver. However, time steps can be larger. Values from previous time step can be taken as initial guess values for current time step.

30

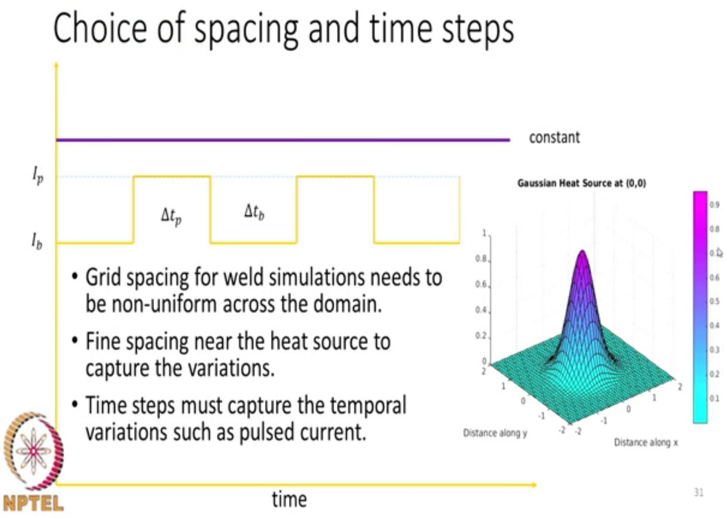
And the reason why that is happening is as follows. Explicit time scheme is used whenever you want the program to be very simple that is the source code writing is a very simple exercise and explicit scheme can give you simple expressions and you do not have any iteration of the loops therefore, the programming part will be quite easy.

However, it is subject to a numerical instability whenever you have the time step that is too large. What is the upper limit of the time step that you can choose is given by Neumann stability criterion that is given here. The smallest time step that you can choose below that you can choose, but the maximum times that you can choose is given as follows square divided by k by rho C p which is basically the thermal diffusivity. Which means basically we are looking at how much time is available for the diffusion of heat across a control-volume of a width delta x and that is a time that you can choose as a maximum time step you can choose something less than that.

And as you can see the delta x it would be very fine at the center of the weld pool where you are going to the heat source, which means that delta x that you are going to use for calculations must be the smallest grid spacing that will be used in the simulation. And the thermal diffusivity is quite large and is going to denominator which means that the delta T is going to be quite small. And this must be also kept in mind whenever you are using further changes in the heat source and we will come to that in a moment.

Implicit scheme would not have any such limit you could use any time step that you wish to use. However, if you use a large time steps you may have to do more iterations. And the previous the time set values can always be taken as initial guess in the case of implicit schemes, whereas explicit scheme you always have the initial condition to start your calculations and so a time marching scheme is available.

(Refer Slide Time: 36:26)



Choice of spacing and time steps

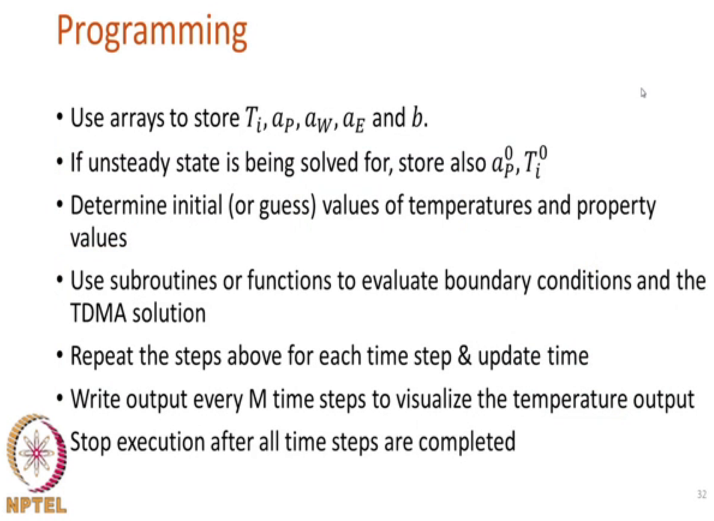
- Grid spacing for weld simulations needs to be non-uniform across the domain.
- Fine spacing near the heat source to capture the variations.
- Time steps must capture the temporal variations such as pulsed current.

Some care has to be taken in the choice of the time step, and this is where we are talking about. For example, let us look at the grid spacing itself. You know that the heat source in the case of welding is going to be uniform; it is going to be focused at a particular location. So, wherever it is being applied they you must have find grid spacing, so that the variation of the heat source is captured properly, which means that the grid spacing should not be uniform across the domain it must be fine at the locations, where the heat source is applied and it can be course away from the heat source.

And when you then use the finest grid spacing to find out according to the Neumann stability criterion, what would be the largest time step that you could choose then you must compare that time step with the pulsing time step which is used in the heat source. Very often the welding is used in a pulsed current mode and then the pulsing is done at a particular frequency and that would also set a particular time step for the change of the heat source. So, the time step you choose in the simulation must be smaller than this pulse.

So, you can say that it could be smaller than either the Neumann stability criterion or the pulse time step whichever smaller than there. It means that the choice of time step is very, very important; otherwise you may actually lose out information with respect to the change of heat source during the pulsing itself. And this pulsing can be as fast as 50 hertz and 50 hertz would mean 20 milliseconds is the time step that you would choose and 20 milliseconds is a very small time if you were to do welding simulations for several seconds or several minutes. And which would mean that the total number of times sets for which you have to calculate would run into several thousand or even tens of thousands.

(Refer Slide Time: 38:14)



Programming

- Use arrays to store T_i, a_P, a_W, a_E and b.
- If unsteady state is being solved for, store also a_P^0, T_i^0
- Determine initial (or guess) values of temperatures and property values
- Use subroutines or functions to evaluate boundary conditions and the TDMA solution
- Repeat the steps above for each time step & update time
- Write output every M time steps to visualize the temperature output
- Stop execution after all time steps are completed

And how do we go about programming, programming exercise is going to involve you to have arrays to store the temperature at various locations. So, you would have one array for the temperature, one array for the coefficient at the p location that a p, one array for the left side coefficient, and one array for the right side coefficient. In case, you use constant grid spacing then you would not need an array to store the coefficients of the linear equation.

However, as I mentioned to you in welding, we normally have non-uniform grid spacing. So, you definitely will have to store the coefficients of the neighboring temperature in arrays. So, you would have as many arrays as the number of neighbors. And then you would also have an array to store the source term which is definitely going to be location dependent.

Essentially you must have a program in which you have multiple arrays to store these variables. And in case, you are going to use unsteady state solution then you need also arrays to store the previous time step values and you normally use the subscript zero to indicate them, and which means that you would have more arrays that would be required if you are going to go for a unsteady state conduction problem.

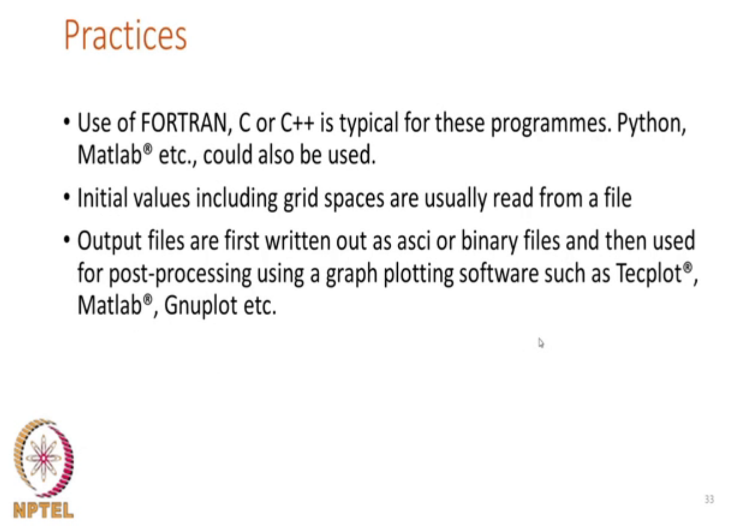
And normally, you would have one sub routine or a function, where you would be setting the initial values of temperature or the guess values of the temperature to start with, and also to set the property values a T various locations. If you have location dependent the properties then you may have to have arrays for the properties also. And if do not have that problem then you can actually use only just the constants that to use at in a program.

In case you are introducing temperature variation a properties, then because temperature is also varying at each location then you may have to again use arrays to store the property values, and you have to update the properties at every time step as and when the temperatures are change.

And then you will have to have separate routines or functions to write the boundary conditions and the tri-diagonal matrix algorithm solution. And you can repeat these solutions at each time step as many time steps as you would need; and after solution, you must copy it to previous time step and then update the time, and then you can keep repeating the cycle as many times as a total amount of time for which the solution is to be done.

And once the time is over then you can write the output data and then use the output data to visualize the temperature field that is calculated, and then you can stop the execution once all the time steps are completed. So, you can see that basically there is a multiplication of the number of operations. So, how big is a problem of computation, you can say that it is as big as the number of time steps as big as the number of grid points and as big as for example, a number of iterations you need to do.

(Refer Slide Time: 41:05)



Practices

- Use of FORTRAN, C or C++ is typical for these programmes. Python, Matlab® etc., could also be used.
- Initial values including grid spaces are usually read from a file
- Output files are first written out as asci or binary files and then used for post-processing using a graph plotting software such as Tecplot®, Matlab®, Gnuplot etc.

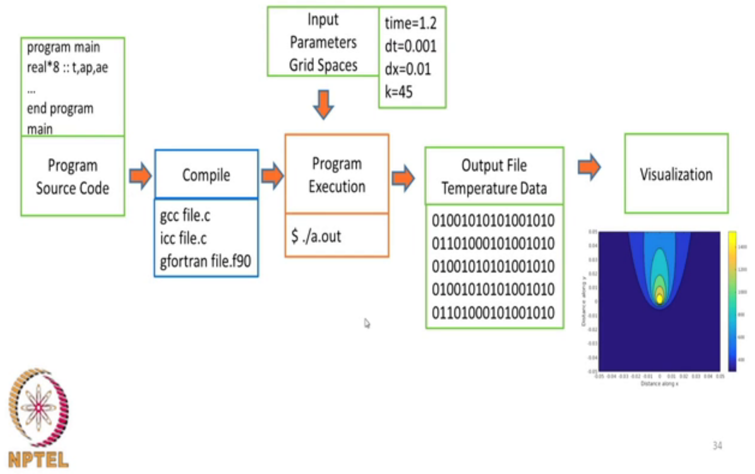
33

And what would one normally do to do these programming very often that the program are written in FORTRAN language, because it is very intuitive to write a mathematical expression directly; you could also use C or C++ languages. (Refer Time: 41:19) Python and MATLAB are also being used, because they can be used for visualization within the same environment. And for 1D problems, surely these are all alternative can be used. And initially values are generally read from a file which means that a program that does a numerical solution should also have the capability to read the input files from a file.

And the grid spaces are also being specified as a file you could have initial values and grid spaces coming as an input into your program. Outputs are then written as either ASCII files or binary files. ASCII files are written when you want to have a look at the output temperature numbers directly yourself; and if you are not interested in that and directly use in them to plot then you can have a faster I O, when it is done with a binary file.

The input output operations are always fast when you use a binary file, which means that when you go to 3D simulation this necessarily you will be using binary input and output because the amount of data will be writing in or reading in would be quite large. And what kinds of tools are used for plotting you could use Tecplot software or MATLAB software or free software such as Gnuplot to use the data and then make the plots out of them.

(Refer Slide Time: 42:36)



```
program main
real*8 :: t,ap,ae
...
end program
main
```

Program Source Code

Compile

```
gcc file.c
icc file.c
gfortran file.f90
```

Input Parameters Grid Spaces

```
time=1.2
dt=0.001
dx=0.01
k=45
```

Program Execution

```
$ ./a.out
```

Output File Temperature Data

```
01001010101001010
01101000101001010
01001010101001010
01001010101001010
01101000101001010
```

Visualization

This is the overall scheme in which we are doing it. So, we have done discretization, and we are in the linearized set of equations, and we are using an algorithm to solve them and then that algorithm is then implemented as a program. And that program is then written in some language such as FORTRAN or C. Once that program is written then you would compile that program using popular compilers you have for example, the GNU compilers, GFortran, GCC or GFortran for example, or you can also use Intel compilers if you are using Intel platform and so on.

Once you compile then an executable is created which can then be run, so that executable in the case of a Unix, Linux environmental will be called as a dot out. And they can if you if you execute that then it would read the input parameters from a file as you have specified, and those parameters can be changed for each execution. And once the program is executed it would be write in the output data and that output data can be a binary format or an ASCII format. And that output data format can be there visualized using software such as Gnuplot or MATLAB. And now you have got the temperature variation available.

So, you can see now how the entire layout is done. So, you start from a differential equation you discretize that; and then you write a linear set of equation, then you find the algorithm to solve them. Then you write a program to implement that algorithm, you compile program, you execute the program and then you write the output data out into a file and then you use that file data file to visualize, and then you can start seeing the temperature variation on the domain as intuitively plotted for example, using a color map for example.

With that, we close the first part this lesson. In the second part, we would take more detailed aspect of the numerical implementation, and we will continue shortly.

Thank you.