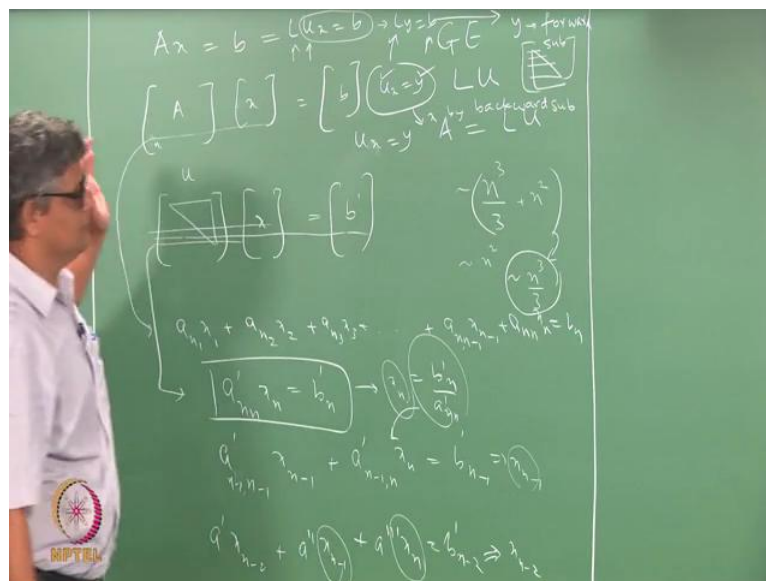


**Computational Fluid Dynamics**  
**Prof. Sreenivas Jayanti**  
**Department of Computer Science and Engineering**  
**Indian Institute of Technology, Madras**

**Lecture – 46**  
**Tri-diagonal Matrix Algorithm: Derivation**

In the last class, we looked at two methods for the solution of  $Ax$  equal to  $b$  type set of simultaneous linear algebraic equations. One of this was a Gaussian elimination method and the other was L u decomposition method. In both cases, we are reducing the set of equations described by  $Ax$  equal to  $b$  into a specific form, into an upper diagonal matrix form, the  $A$  is converted into an upper diagonal matrix form. In the process, we are rearranging the equations, reformulating the equations in such a way, that solution by successive substitution is possible.

(Refer Slide Time: 01:06)



In the case of Gaussian elimination method, we are converting  $Ax$  equal to  $b$  into, we are converting  $Ax$  equal to  $b$  into a special form, which is being written as an upper, train, triangular matrix;  $u$  times  $x$  here equal to a modified form of the coefficients. And this triangular, upper triangular form is such, that while the last equation here contains  $A$ , the last equation here is a  $n-1 \times n-1$  plus a  $n-2 \times n-2$  plus a  $n-3 \times n-3$  plus a  $n \times n$  minus  $1 \times n$  minus

1 plus  $a_{nn} x_n$  equal to  $b_n$ .

So, this is rewritten in this modified form. The last equation is written as  $a_{n,n-1} x_{n-1}$  equal to  $b_n - a_{nn} x_n$ . And the one which is just above this, this equation here is written as  $a_{n,n-2} x_{n-2}$  plus  $a_{n,n-1} x_{n-1}$  equal to  $b_n - a_{nn} x_n$ . So, this is  $a_{n,n-2} x_{n-2}$  plus  $a_{n,n-1} x_{n-1}$  equal to  $b_n - a_{nn} x_n$ .

So, in this transformed rearranged set of equations, the last equation is simplified from one, which involves all the variables, all the way from  $x_1, x_2, x_3$ , up to  $x_n$  into an equation involving only  $x_n$  as a variable. So, from this we can directly get  $x_n$  as  $(b_n - a_{nn} x_n) / a_{nn}$ . And once you have  $x_n$  here, you can substitute this value in this and you have this equation, which contains only  $x_{n-1}$ . So, we can, by substituting this value in this we can get  $x_{n-1}$ .

Similarly, the 3rd equation from bottom here is put in some form of  $x_{n-2}$  plus a double prime  $x_{n-1}$  plus a triple prime  $x_n$  equal to  $b_n - a_{nn} x_n$ . So, the right hand side of this equation gets changed and the coefficients also gets changed, but by this time we already know  $x_n$  and we already know  $x_{n-1}$ .

So, we can substitute here for this and we can substitute here for this and we have a simple equation for  $x_{n-2}$ , we can get from this  $x_{n-2}$ . So, by successive back substitution, starting from the bottom most equation, the last equation, all the way up to this, we can evaluate  $x_n, x_{n-1}, x_{n-2}$ , all the way to  $x_1$  through substitution alone, which is very fast. So, it would not take large number of equation, large number of a multiplications and divisions.

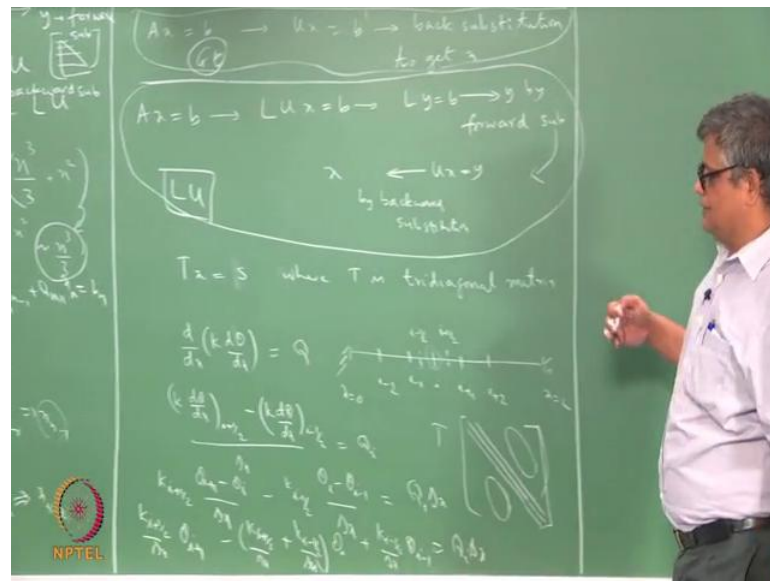
For example, here we have one division and here we have these times this multiplication and then again subtraction of this from this and then division by this. So, you have one multiplication here and one division here. So, and here we have one multiplication, one multiplication, subtraction and division here. So, as we go more and more, we get more and more number of mathematical operations, but it is only every time a few more. So, the back substitution process is an easy way of solving the equations successively, but to go from a general matrix into an upper triangular matrix here requires a lot of work.

And we said, that that will be taking  $n^3$  by 3 number of mathematical operations, whereas the back substitution will take  $n^2$  number of operations and together it will take  $n^3$  plus  $n^2$  number of mathematical, arithmetic operations, which will tend towards  $n^3$  by 3 number of multiplications and divisions for the solution using Gaussian elimination method and which is much, much better than the factorial  $n + 1$ , which is required for the Cramer's rule.

The other method that we looked at, which is the L-u decomposition method is similar to this in principle. In this case, in the case of Gaussian elimination, as we transformed  $Ax = b$  into  $Ux = b'$ , here the right hand side is also getting transformed. In the case of L-u decomposition, we only convert  $A$  into a product of lower and upper triangular matrix so that an equation of  $Ax = b$  is written as  $L U x = b$  in which we know we have certain algorithms to find the elements  $L_{ij}$  and  $U_{ij}$  of this lower and upper triangular matrices.

So, when you have an equation like this, then you can put this, you can solve this equation first. So, that is, you can solve, you can, you can put  $Ux = y$  so that we write this as  $Ly = b$ . So, in this  $L$  is known and  $b$  is known, so we can get  $y$ . And this lower triangular matrix will be like, this matrix is like, here, so we get  $y$  by starting with the 1st equation and 2nd equation, 3rd equation like this. So, this we get by forward substitution. And once we get  $y$  here we can, we can solve for  $Ux = y$  in which case  $y$  is known and  $U$  is known, from this we get  $x$  by backward substitution.

(Refer Slide Time: 08:27)



So, let us just try to write it down neatly here. We have  $Ax$  equal to  $b$  converted to  $u x$  equal to  $b$  prime and solve by back substitution. The first method is the Gaussian elimination method. The 2nd method,  $A x$  equal to  $b$  written as  $L u x$  equal to  $b$ , the same  $b$ , and solving this in two steps as  $L y$  equal to  $b$  giving  $y$  by forward substitution, and this is followed by solution of  $u x$  equal to  $y$  and solving this for  $x$  by backward substitution. So, this is the essence of the L-u decomposition method, and this is what we are doing in Gaussian elimination method. Both of them are good methods, in the, for the general case of  $Ax$  equal to  $b$ .

But today we are going to look at a specialized method which builds on the same principles, that if it is possible to convert  $A x$  equal to  $b$  into a special form into an upper triangular form or  $L u x$  equal to  $b$  form like this, then we can get easy solution through successive substitution here.

The same principle can be applied to a simple case of a tridiagonal matrix in a case where we have  $T x$  equal to  $s$ , where  $T$  is tridiagonal matrix. So, let us see, for example, a simple case of  $d$  by  $d$  x of  $K d t$  by  $d$  x. Let me put, let me not confuse this  $t$  with this  $T$ , so let me put this as  $\theta$  equal to  $Q$ . So, this is one-dimensional steady state heat conduction equation where  $K$  is the thermal conductivity,  $\theta$  is the temperature and  $Q$

is some heat source or heat sink in the appropriate form, for example, volumetric or area wise and all that.

So, we have this equation and we have this is the one-dimensional problem. We have  $x$  going in this direction and we can have  $i$  here,  $i - 1$  and  $i + 1$ ,  $i + 2$ ,  $i - 2$ , like this. The boundary conditions at  $x$  equal to 0 and  $x$  equal to  $L$  are specified. At this point we can discretize this by writing this as, we will put this as  $i - \frac{1}{2}$  and this as  $i + \frac{1}{2}$ , that is, midway between the two points. So, we can write this as  $K \frac{d\theta}{dx}$  at  $i + \frac{1}{2}$  minus  $K \frac{d\theta}{dx}$  at  $i - \frac{1}{2}$  divided by  $\Delta x$  is equal to  $Q_i$ . And we can write  $\frac{d\theta}{dx}$  at  $i + \frac{1}{2}$  here as  $\frac{\theta_{i+1} - \theta_i}{\Delta x}$  and this as  $\frac{\theta_i - \theta_{i-1}}{\Delta x}$  is equal to, we take this here and we have  $Q_i \Delta x$ .

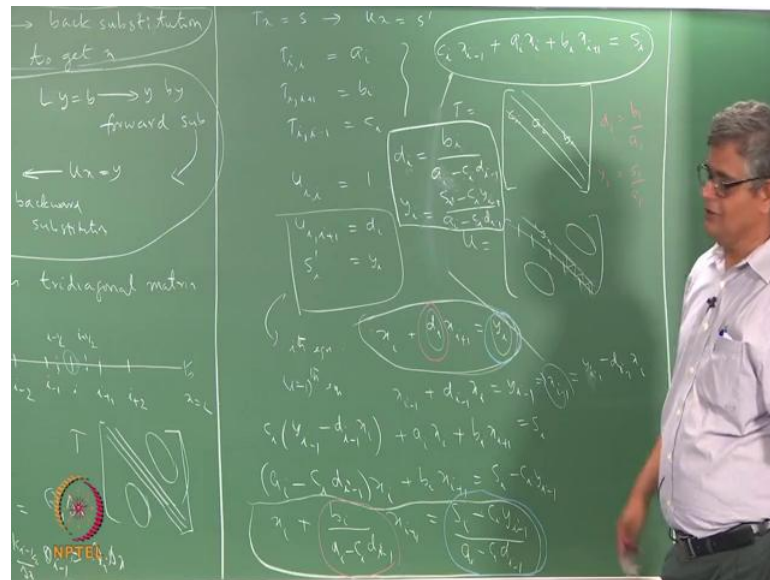
So, here we see, that we have  $\theta_i$ ,  $\theta_{i+1}$  and  $\theta_{i-1}$ . So, we can write this as  $K \frac{\theta_{i+1} - \theta_i}{\Delta x} - K \frac{\theta_i - \theta_{i-1}}{\Delta x} = Q_i \Delta x$ .

So, we have  $\theta_i$  in terms of  $\theta_{i+1}$  and  $\theta_{i-1}$ . So, if you do this for every  $\theta_i$  and put them in the lexicographic, graphic ordering, we will get a matrix  $A$ , which will have all  $\theta_i$  values will be along the central diagonal here and  $\theta_{i-1}$  will be just here and  $\theta_{i+1}$  will be here, all these are 0s. So, this  $A$  resulting from the discretization of one-dimensional heat conduction equation is a typical example where we have  $T$  as a tridiagonal matrix. So, this is the tridiagonal matrix form of the coefficient matrix  $T$ , which arises from this one dimensional Poisson equation or Laplace equation like this.

So, what we would like to do is that we would like to solve this and we would like to solve this taking advantage of this type of conversion, that is, conversion into, into an upper triangular matrix or L-u kind of thing here. So, we, it is possible to develop a very fast scheme by taking advantage of a, the fact, that in this there is only 3 rows, 3 diagonals here, which are non-zero and all the others are 0 and taking advantage of the fact, that in a typical diffusion problem, the diagonal dominance of this triangular matrix

is satisfied.

(Refer Slide Time: 16:11)



So, with these kind of conditions what we would like to do is to convert  $T x$  equal to  $s$  into an upper triangular matrix of the form  $u x$  equal to  $s$  prime, okay and in such a way, that this upper triangular matrix will only have 2 rows. So, what we are looking at is, that where you have  $T$  such that the  $T_{i,i}$  is  $a_i$  and  $T_{i,i+1}$  is  $b_i$  and  $T_{i,i-1}$  is  $c_i$  where  $a, b, c$  are the coefficients. So, that for the  $i$ th equation in this, so you have 1st equation, 2nd equation, 3rd equation, all the way up to  $n$  number of equations, the  $i$ th equation can be written as  $c_i x_{i-1} + a_i x_i + b_i x_{i+1}$  equal to  $s_i$ . So, the  $i$ th equation in this system is written as a coefficient corresponding to  $i-1$ , a coefficient  $a_i$  corresponding to  $i$ th term and coefficient  $b_i$  corresponding to the  $i+1$ th variable equal to  $s_i$ .

And this formulation is such that each of these coefficients can change from equation to equation. You can have some special cases where all the  $c$ 's are constant, all  $a$ 's are constant. For example, in the simplest case where  $K$  is constant, the thermal conductivity is constant; we normally have  $c$  equal to 1,  $a$  equal to minus 2 and  $b$  equal to 1. So, this represents a tridiagonal, an equation, a linear algebraic equation involving 3 immediate, 3 adjacent neighbors. And when you go for all values of  $i$ , you get this type of form. So,

this is an equation, that we wish to solve for  $n$  number of  $x_n, x_{n-1}, x_{n-2}, x_{n-3}$ , all the way up to  $x_n$  and we would like to convert this into  $u x = s$ , where  $u_i$  is taken to be 1 and  $u_{i+1}$  is given as  $d_i$  and  $s_i$  is given as  $y_i$ .

So, the form of  $T$  is this. Central diagonal has coefficient  $a_i$ , the one corresponding to  $i, i+1$  is the one which is above here. All the coefficients here are  $b_i$  and the one  $i, i-1$  is one below and this is  $c_i$ . So, these are the coefficient values and these values may change in each equation, as we mentioned here. This is being converted into a special form of upper triangular matrix where we have, all the diagonals have 1 and we have only one more diagonal here and this is  $y_i$ , and all these are 0. So, if you have, if you can make this conversion here, then this equation can be solved in the back substitution process. We have this one directly giving as  $x_n$  and this one will involve  $x_n$  and  $x_{n-1}$ . We know  $x_n$ , so we can get  $x_{n-1}$ .

We go to the next equation. The next equation was only  $x_{n-1}$  and  $x_{n-2}$ . We solve for  $x_{n-2}$ , then you come here, this involve  $x_{n-2}$  and  $x_{n-3}$  and so, we, we know  $x_{n-2}$ . So, we can solve this for  $x_{n-3}$  we can just climb the ladder, as it were, you solve this and then you go here and then get this and then you go here, get this, you go here, get this, like this and you can go all the way up to the first one. So, this is the idea of the, tri, tridiagonal matrix algorithm in which tridiagonal original matrix  $T$  is converted into an upper triangular matrix containing only two non-zero diagonals.

So, how can we do this conversion is the thing that we would like to have here. So, when we write an equation like this, this equation is such that the  $i$ th equation is written as  $x_i + d_i x_{i+1} = y_i$ . So, because  $u_i$  is equal to 1, so in the  $i$ th equation you have  $x_i$  and  $x_{i+1}$ ;  $x_{i+1}$  will be coming here. So, this is the  $i$ th equation in  $u x = s$ ,  $u x = s$  and this is the  $i$ th equation in  $T x = s$ . So, whereas, here you have three coefficients  $c_i, a_i, b_i$  here. In this equation, this coefficient is 1 by choice and this coefficient,  $d_i$  and  $y_i$  are to be determined yet. So, if this is the  $y$ th equation,  $i-1$ th equation is given by  $x_{i-1} + d_{i-1} x_i = y_{i-1}$ .

So, now what we like to do is to convert this equation in which we already know  $c_i$ ,  $a_i$ ,  $b_i$ ,  $s_i$  because that is what we have derived here. So, all the coefficients in this T are known, we want to convert this into this form and determine this  $d_i$  and  $y_i$  for each of these equations because once we do this, then it is possible to do this back substitution here. So, we make use of this expression here in this. So, we can write this as. So, we can say from here, this  $x_i - 1$  is  $y_i - 1 - d_i - 1 x_i$  here. We substitute this here and then get rid of this  $x_i - 1$ , so that we have an equation involving  $x_i$  and  $x_i + 1$  and that equation is similar to this form. This is also an equation involving  $x_i$  and  $x_i + 1$ .

So, we compare the coefficients and then, from that we will be able to form an equation for  $d_i$  and  $y_i$ . So, what we are doing is, if this is the form of  $i$ th equation, we are saying, that form of the  $i$ th equation is  $x_i - 1 + d_i - 1 x_i = y_i - 1$ . So, this equation allows us to write  $x_i - 1 = \text{this} - \text{this}$ . We substitute this value here in this and we can write that as  $c_i \text{ times } y_i - 1 - d_i - 1 x_i + a_i x_i + b_i x_i + 1 = s_i$ .

So, we can now take this  $c_i - 1$  here and we have  $a_i - c_i d_i - 1$  times  $x_i + b_i x_i + 1 = s_i - c_i y_i - 1$ . So, we can divide this whole thing by this and write this as  $x_i + b_i$  by  $a_i - c_i d_i - 1$  times  $x_i + 1 = s_i - c_i y_i - 1$  divided by  $a_i - c_i d_i - 1$ . So, I would like you to examine this equation and this equation here. In both cases, this coefficient is 1 and this coefficient must be equal to this coefficient. So, from this we can say, that that this term and this is equal to this and we have  $y_i$  is given by this.

So, let us now write, the formula here  $d_i$  is  $b_i$  by  $c_i a_i - c_i d_i - 1$  and  $y_i$  is  $s_i - c_i y_i - 1$  divided by  $a_i - c_i d_i - 1$ . So, that means, that we already know  $a_i$ ,  $b_i$ ,  $c_i$  everywhere and  $s_i$ . We do not know  $d_i - 1$  and  $s_i - 1$ , but we can say, that for the  $d_1$  is  $b_1$  by  $a_1$  and  $y_1$  is  $s_1$  by  $a_1$ . So, starting with these expressions, from  $i = 1$  equal to next point onwards, that is, we can take  $i = 2$  onwards, we can put  $d_2 = d_1$ , which we know  $a_2$ , which we know  $c_2$  times  $d_1 - 1 = d_1$  and  $d_1$  is already known, so we can evaluate  $d_2$ . Similarly,  $y_2$  will be equal to  $s_2$ , which we know,  $c_2$  times  $y_1$  which is given here. And  $a_1$  is known,  $a_2$  is known,  $c_2$  is known

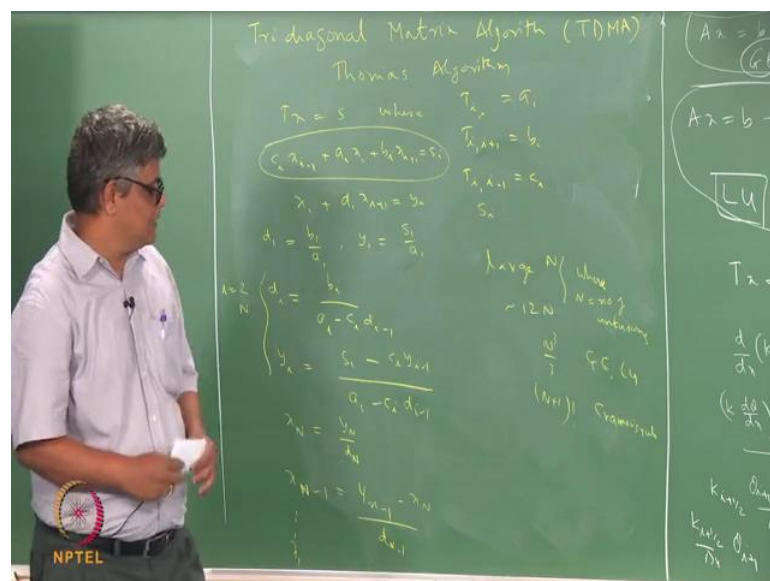


and  $d_1$  is known, so we can evaluate  $y_2$ .

So, we can evaluate  $d_2, y_2$  and then, we go back here and when we evaluate  $d_3, y_3$  and  $d_4, y_4$  all the way up to  $d_n, y_n$ . So, using this recursive formulas, starting with known values of  $d_1$  and  $y_1$  from the form of these equations we will be able to get the six, determine all the coefficients of  $d_i$  and  $y_i$ . Then, it is a question of back substitution. So, we will write down the whole algorithm for the solution of  $Tx = s$  using this tridiagonal matrix algorithm in which we write  $Tx = s$  in the form of an upper, tri, tridiagonal, diagonal, upper diagonal matrix, bidiagonal matrix involving only two diagonals in which we put  $u$  equal to 1; here all these diagonal elements to be 1.

Why we want to put these to be 1? These can be any values, but this had to be fixed in order to get unique set of values here. So, in this algorithm we put all of them to be 1 and then, we have exactly as many number of equations as are needed to determine this  $d_i$  and  $y_i$  and this can be put in the simple recursive forms. So, we, from known values of  $a_i, b_i, c_i, s_i$ , we start with the initial values and then, we find all the  $d_n$  and  $y_n$  and through the back substitution process we get the final solution.

(Refer Slide Time: 30:51)



So, we will just rub it. I am continuing the tridiagonal matrix algorithm known as

TDMA. It is also known as the Thomas algorithm. So, the solution of  $Tx = s$ , where  $T_{ii}$  is  $a_i$ ,  $T_{i,i+1}$  is  $b_i$ ,  $T_{i,i-1}$  is  $c_i$  and  $s_i$  is  $s_i$ . So, with this, starts with conversion of this, this equation, that is  $c_i x_{i-1} + a_i x_i + b_i x_{i+1} = s_i$ . So, this is rewritten as  $x_i + d_i x_{i+1} = y_i$  where  $d_i = b_i / a_i$  and  $y_i = s_i / a_i$  and we have  $d_i$  is given by  $b_i / a_i$  and  $y_i$  is  $s_i / a_i$ .

So, we start with this and then we evaluate all this from  $i = 2$  to  $n$ . And once you have this, you can then back substitute by saying  $x_n = y_n / d_n$  and then, we can write  $x_{n-1} = (y_{n-1} - d_{n-1} x_n) / a_{n-1}$  and so on. We can go through this; we can go through this until we go from  $x_n$  to  $x_1$ , all the way to  $x_1$ .

So, this is the Thomas algorithm where we convert the  $Tx = s$  into  $Ux = s'$  where  $U$  is an upper bi-diagonal matrix and then we evaluate the elements here and then, we do back substitution to get this. So, the overall number of mathematical operations required for the solution for large  $n$  is about  $12n$  where  $n$  is the number of equations. So, this is very small compared to the  $n^3$ , which is required for the Gaussian elimination and L-u method and  $n!$  for Cramer's rule. So, but the advantage is, that the disadvantage is this can be applied only for tridiagonal matrix, which can be applied only for one-d problems.

We will see later on in the next class some other methods and we will come back to this method in, in the advanced methods, which try to make use of efficiency of this Thomas algorithm in solving one particular form of  $Ax = b$ .

Thank you