Computer Aided Power System Analysis
Prof. Biswarup Das
Department of Electrical Engineering
Indian Institute of Technology-Roorkee

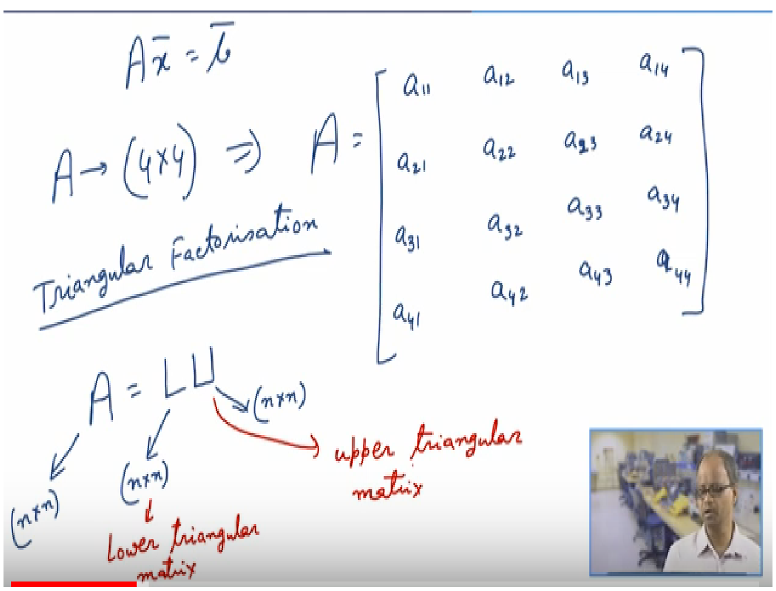
Lecture - 30
Triangular Factorization

Welcome to this lecture on course computer aided power system analysis. We have been now discussing the procedure for solving a set of linear equations without involving explicit matrix inversion. In that process for the last few lectures we have been looking into the process of Gaussian elimination and we have also looked into the problem of fill-in process fill-in problem which can occur in the Gaussian elimination method.

And we have also looked into the case where a simple case of optimal ordering can to some extent address the problem of the fill-in process. Now from today onwards we would be looking into another very powerful method of solving a set of linear equations without involving explicit matrix inversion and that method is called triangular factorization method.

This method is also known as more popularly as LU factorization method. So now let us start looking into this in more detail.

(Refer Slide Time: 01:50)



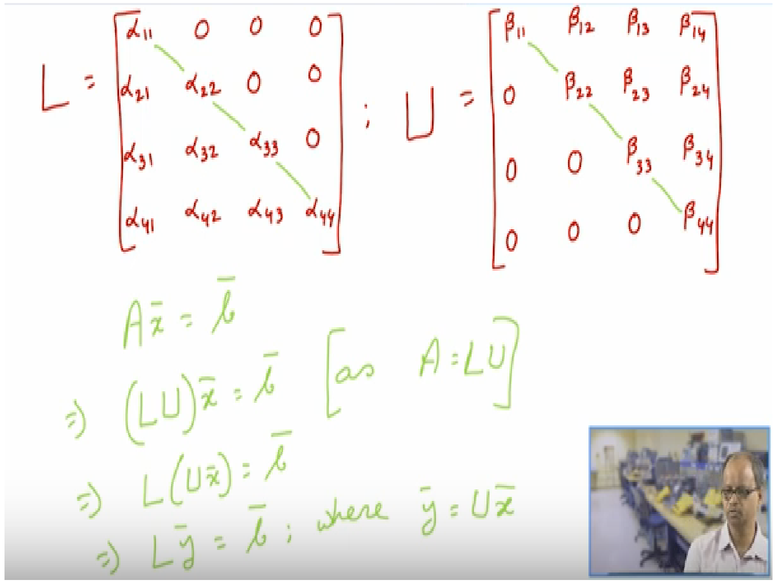
So now actually what we have is here is that we have an equation that A x = b where x is the vector of unknown quantities and b is the known quantities and A are the coefficient matrix and we like to solve for this vector x. Now let us say just as an example, let us say that we have an A is a let us say 4 * 4 matrix let us say. So let us say that A is given by a 11, a 12.

We are trying to show this method by an example and then after that we will immediately understand that this method is very easy to extend for any general N * N matrix. But for the purpose of understanding we are taking a very typical example. So this is a 4 * 4 matrix. Now what we do in LU factorization or triangular factorization, when we do triangular factorization what we do is that we have a matrix A.

We do represent this matrix as the product of 2 matrix L and U. So this is the original matrix. Let us say this is N * N. So this is an N * N. For example here N is 4. This is also an N * N matrix. This is also an N * N matrix. But this matrix is a lower triangular matrix. And this is, this U is an upper triangular matrix. So what is meant by a lower triangular matrix?

As we know that a lower triangular matrix is a matrix where all the elements at and below the main diagonals are nonzero and all the elements above the main diagonal are 0. Similarly, an upper triangular matrix is a matrix where all the elements below the main diagonals are 0. So then therefore if we, so then therefore in our example as A is a 4 * 4 matrix,

(Refer Slide Time: 04:57)



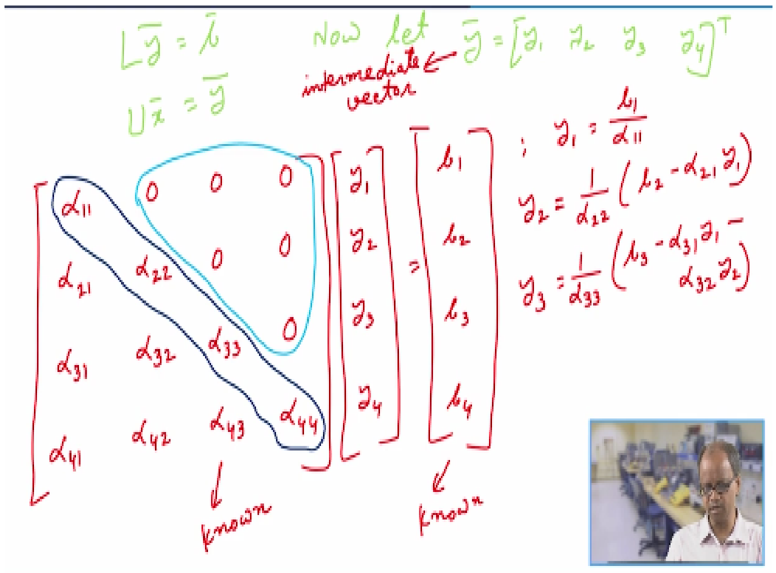
So then therefore for this matrix, for this 4 * 4 matrix L would be something like this. So L would be something like let us say alpha 11. Then 0, 0, 0; alpha 21, alpha 22. Then alpha 31, alpha 32, alpha 33, and alpha 41, alpha 42, alpha 43, alpha 44 and all the elements are 0. So this is an lower triangular matrix where all the elements above the main triangle are 0. So it is a lower triangular matrix. What is our main diagonal?

This main diagonal is alpha 11, alpha 22, alpha 33, alpha 44. Similarly, U is an upper triangular matrix. So then if U is an upper triangular matrix, so then therefore what will happen is that have let us beta 11, then beta 12, beta 13, beta 14. Then we have 0, beta 22, beta 23, beta 24. Then we have 0, 0, beta 33, beta 34. Then we have 0, 0, 0, and beta 44. So here also all the elements below the main diagonal is 0.

So here in this case the main diagonal is nothing but this one. So this is the main diagonal, this is the main diagonal. And here also is the main diagonal. Here is also main diagonal. So this is upper triangular matrix and this is the lower triangular matrix. So if we have this upper triangular matrix and lower triangular matrix so then how does it help to solve us this unknown quantity x. So let us see. So what we have?

We have got the original equation is A x = b. Now A is LU. So then therefore (LU) x = b [as A = LU]. Now this we can write down as L(Ux) = b. Please note that this is N * N, this is N * N and this is N * 1. So then therefore if we even do this so this matrix order does not change. So then I can write down that Ly = b where y = Ux, right? So what we have?. So now we have got 2 equations.

(Refer Slide Time: 08:49)



One equation is Ly = b and another equation is U x = y. So then therefore if we know this vector y we can calculate this vector x. Now let us see, now **if** because Ly = b so then let us say that now let y vector is let us say y 1, y 2, y 3, y4. We are just taking the case of this our specific 4 * 4 system. So then what we have is, so if we just write this equation alpha 11, 0, 0, 0; alpha 12, alpha sorry alpha 21; alpha 21, 0, 0.

Then alpha 31, alpha 32, alpha 33, 0 and alpha 41, alpha 42, alpha 43, alpha 44; [y 1, y 2, y 3, y 4] = b 1, b 2, b 3, b4. So this becomes the equation. So set of equations. Now here let us assume that we know that how to decompose A = LU. Of course we will be looking into an algorithm to decompose A = LU. At present let us assume that we know that what this matrices L and U are.

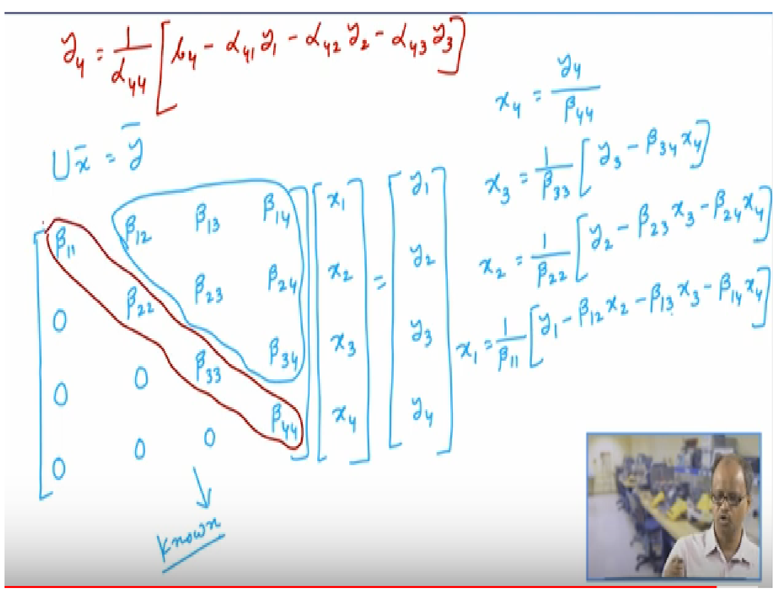
That is basically for a given matrix A we know that what this matrices L and U are. So then therefore once we know this matrix L and U so now this matrix is known matrix. So then therefore once we know this matrix L so then these are known matrix. So this is also known. So now we can solve for y 1, y 2, y 3, y 4. So now here what will be y 1; y 1 would be simply b 1/alpha 11.

What would be y 2; y 2 would be simply 1/alpha 22 * b 2 – alpha 21 y 1. So once we know y 1 we will simply plug it here; b 2 is known. All this alphas are known. Similarly, y 3 would be 1/alpha 33 * b 3 – alpha 31 y 1 – alpha 32 y 2 and lastly y 4 would be simply 1/alpha 44 = b 4 - this * this - this * this - this * this, right? And lastly y 44 would be y 4 would be 1/alpha 44 [b 4 – alpha 41 y 1 – alpha 42 y 2 – alpha 43 y 3].

So once we know this, so by this very simple process, just algebraic manipulation we would be able to find out this quantities y 1, y 2, y 3, and y 4. So from the known values of b 1, b 2, b 3, b 4 and this known coefficients alpha's coefficients that is this known L matrix we would be able to calculate the intermediate vector y 1, y 2, y 3, y 4. So y is basically it is called an intermediate vector.

Now once we know this intermediate vector, then we would be able to calculate this unknown quantity x.

(Refer Slide Time: 13:27)



Now our equation is U x = y. Now what is U? U is beta 11, beta 12, beta 13, beta 14. Then it would be 0, beta 22, beta 23, beta 24. It would be 0, 0, beta 33, beta 34. Then it would be 0, 0, 0, beta 44. So this = x 1, x 2, x 3, x 4. That is equal to this intermediate vector y 1, y 2, y 3, y 4. So we know y 1, y 2, y 3, y 4. We have just now calculated them. And this matrix, U matrix is known.

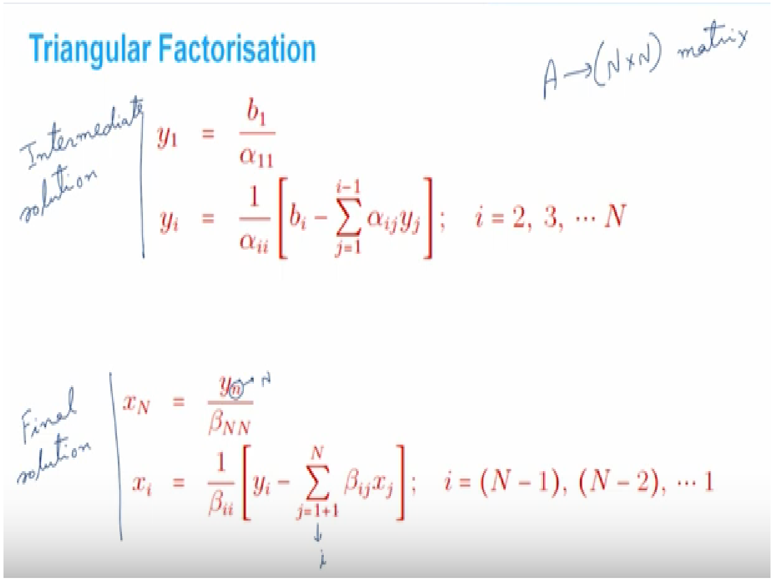
So then from here we can solve for x 1, x 2, x 3, x 4. But however, right now we have to start from the bottom. So we will first solve for x 4. So that is = y 4/beta 44. Then we solved for x 3 is 1/beta 33 [y 3 – beta 34 x 4]; x 2 = 1/beta 22 [y 2 – beta 23 x 3 – beta 24 x 4] and the last x 1 would be 1/beta 11 [y 1 – beta 12 x 2 – beta 13 x 3 – beta 14 x 4]. So by this method, so now by this we have got all this.

Again here also we do not need to invert this matrix U. in the previous step also we did not need to invert this matrix L. So by simple algebraic manipulation we are being able to calculate this unknown quantities x 1, x 2, x 3, x 4. So then therefore as we can see this is a very powerful method in a sense that here also we do not need any matrix inversion. Rather we do not need an explicit matrix inversion.

But what we do that we simply represent the original matrix A as the product of 2 triangular matrices. One is basically a lower triangular matrix and another is an upper triangular matrix and then after that by simple algebraic manipulation we are able to find out the unknown vector x. So then here after we know this LU factorization of a given matrix so then after that this solution is very easy.

So now let us look at the, so here we have simply looked into a very specific case of 4 * 4 matrix just to understand the basic process. But in the case of N * N matrix and especially when we would like to code this basic process so then we need to develop some general expression. So let us look at that particular general expression. So in that general expression, so first we are simply talking about,

(Refer Slide Time: 17:56)



So this part is nothing but this intermediate vector. So we can see that y, so here we are basically, here this is nothing but the intermediate. So this is the intermediate vector. And this is the final solution. So this is the final solution. So this is the intermediate solution and this is the final solution. So in the intermediate solution as we have already seen y 1 would be = b 1/alpha 11 and that is precisely what we have seen.

You see y 1 would be b 1 = alpha 11 and then y i for 2, 3, N where N is the number of unknown. So here we are talking about an N * N matrix. So A is an N * N matrix. So y 2 would be 1/alpha 22 when i = 2 into b 2 – j would be varying from 1 to i – 1. So then when i is 2 it would be j = 1 to 1. So then it will be just j = 1. And it would be alpha 21 * i 1 and that is precisely what it is; y 2 is b 2 – alpha 21 * y 1 and 1/alpha 22.

Then just crosscheck what would be y 3; y 3 would be 1/alpha 33 * b 3. Now here j would be varying from 1 to 2 because i = 3, i -1 = 2. So then j would take 2 values 1 and 2. So it would be b 3 – alpha 31 y 1 – alpha 32 y 2. So it would be b 3 – alpha 31 y 1 – alpha 32 y 2. So this is exactly it is 1/alpha 33 b 3 – alpha 31 y 1 – alpha 32 y 2. Similarly y 4 would be something like this. So this is the general expression.

Now after we get this intermediate vector y, then we have to calculate this final solution x N. Now in the case of final solution, what we do is you see if we look at this we first actually start from the last component. That is if you have got vector x and that vector x has got N components x 1, x 2, x 3, x 4 up to x N. So then we start actually from the last component x N. For example here we have taken an example of 4 * 4 matrix.

So then therefore N = 4. So we have started with x 4 and that x 4 was given by y 4/beta 44 and here it is exactly that; x N = y N/beta NN A. I mean it should be actually capital N. So I must change this. So this should be capital N. So x N would be y N/beta NN and then i will start varying from N – 1, N – 2 up to 1. For example here once we have solved for 4 last one then we have solved for 3 to 1.

So we actually start from the last one and then come to the first one. So this is N, N – 1, N – 2 and 1. So then similarly here also after this last one is solved then I will take the value N – 1. So it would be 1/beta N – 1 N -1. For example N was 4. So it would be 1/beta 33 in case. So x 3 would be beta 33; y 3 because i is taking a value of N – 1. So y 3 and here j is equal to this is actually i + 1. This is actually, this is a printing mistake.

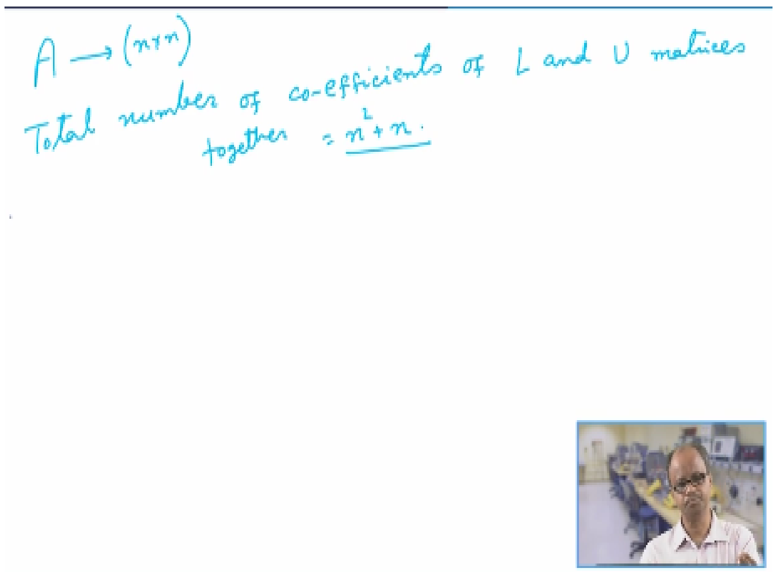
This is actually i + 1. So then from here what we get is 1/beta 33 y 3. So then here we get Y (N – 1) – beta 34 x 4 and here beta i is 3. This is N – 1 and j is actually it is basically i – 1 to N to N. Now here it was actually 3 to 4 to 4. For example here when I have started in this case N is 4. So i is 3. So it would be if i is 3 so it is 3 + 1 4 and N is 4. So j would only take only one value. It is 4 to 4. So it would be beta 34 * x 4.

So it would be beta 34 * x 4. So then that is what precisely it is. So it is – beta 34 * x 4. Similarly, when i takes N – 2 that is i becomes 2. So it becomes 1/beta 22 * y 2 minus. Now i is 2 N is 4 so then j will take the value 3 and 4. So it is 2 + 3 that is 3 and 4. So it would be beta i is 2 and j takes 3 and 4. So beta 23 x 3 – beta 24 x 4. So it would be y 2 beta 23 x 3 – beta 24 x 4.

And the last one when i = 1 so it is 1/beta 11 y i N and j takes the value because i is 1 and i + 1. So it is 2 3 4 N = 4 in our case. So it would be beta 12 * x 2 – beta 13 * x 3 – beta 14 * x 4. And that is precisely what it is; x 1 = 1/beta 11 * y 1 – beta 12 * x 2 – beta 13 * x 3 – beta 14 * x 4. So then therefore these expressions are pretty general expressions. So once we get this matrix L and U from this original matrix A so then after that just by applying this formula and these are formulas very easy to code, it hardly takes any.

So this formula very easy to code. So just by applying this formula we can simply solve the unknown quantities x 1, x 2, x 3, and x 4. Now the question is, so now we come to the most crucial question that how to get or how to obtain this matrices L and U from a given matrix A. So to understand that first let us look at a very simple thing.

(Refer Slide Time: 24:58)



A is an n * n. Now let us look from, and L is also an n * n matrix u is also an n * n matrix. For example in our case when L is an n * n matrix in our case how many elements are there in L? 1, 2, 3, 4, 5, 6, 7, 8, 9, 10; 10 elements. Similarly, in U also we have got 1, 2, 3, 4, 5, 6, 7, 8, 9, 10; 10 elements. So then therefore we have to solve for 10 + 10 that is 20 elements. But now what happens here?

For an n * n matrix, the total number of unknown to solve L and U is total number of unknown, total number of coefficients of L and U matrix together is n square + n. Why it is n square + n? For example it will be very apparent. If we just look at this L so we have got this and if we just let us say superimpose this U matrix over this L matrix what we will have? These elements would be superimposed here.

And on top of that these 4 elements, these 4 elements beta 1, beta 2, beta 3 and beta 4, these 4 elements would be actually superimposed on these 4 elements. So these would be superimposed on these 4 elements. So then therefore it is a basically full matrix plus extra 4 unknowns. That is this extra 4 unknowns would be corresponding to this diagonal matrices. Because this diagonal elements are coming twice.

So then therefore the total number of unknowns, unknown coefficient to be solved for L and U matrix is n square + n. But the number of equations we have got only n square. So then therefore we have to look into some algorithm how to solve for this n square + n unknowns from this equations n. So we would look into this algorithm of LU factorization in the next lecture in more detail. Thank you.