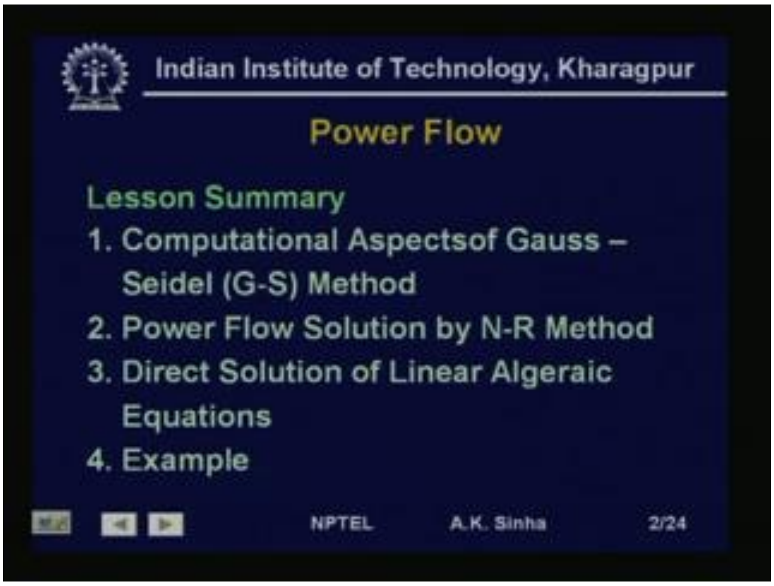
Power System Analysis

Prof. A. H. Sinha
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

Lecture - 18
Power Flow – III

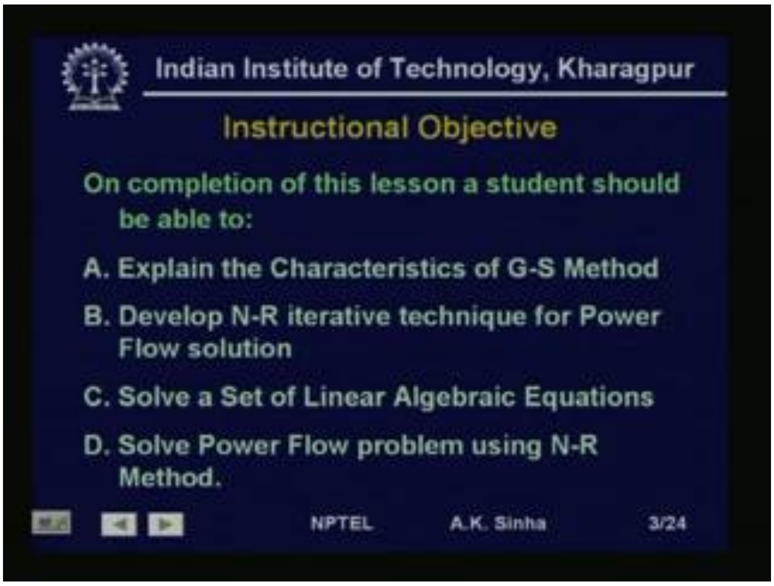
Welcome to lesson 18, in Power System Analysis. In this lesson, we will continue with our dicussion of Power Flow soloution methods. In the last lesson, we discussed about the Gauss Siedel method for power flow solution. In this we will continue with some of those aspects. And then we will talk about newer method for power flow solution.

(Refer Slide Time: 01:21)



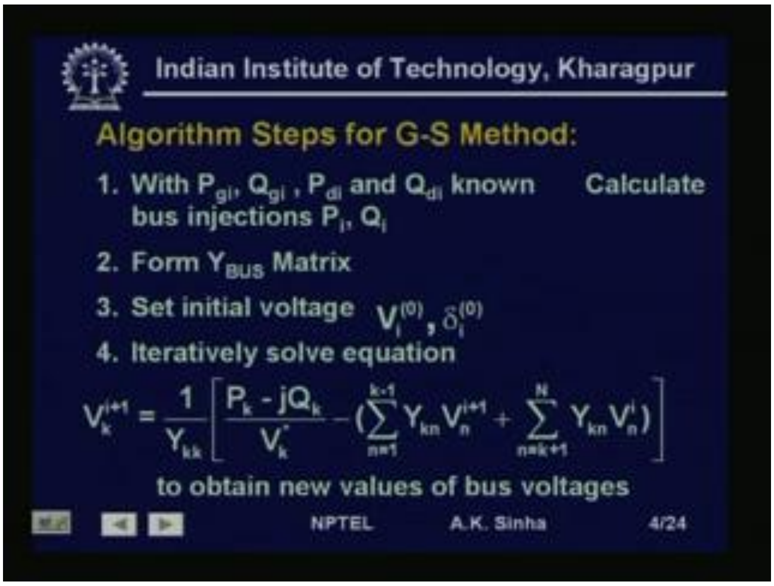
Well, in this lesson, we will start with from computational aspects of Gauss Seidel method. Then, we will talk about power flow solution by Newton-Raphson method. We will, then talk about direct solution of linear algebraic equation, as we will find that, Newton Raphson method, finally ((Refer Time:01:46)) down. To solving, mean a set of linear alegraic equations. And then we will take a small example of solving this linear algebraic equation by Gauss elimination method.

(Refer Slide Time: 02:07)



Well, the objective this lesson, that at the end of this lesson. You should be able to explain the characteristics of the Gauss Siedel method. Develop Newton Raphson iterative technique, for power flow solution. You should be able to solve a set of linear algebraic equation. And solve power flow problem using Newton Raphson method. Of course, this part we will take up in the next lesson, as well as we will take up some example.

(Refer Slide Time: 02:44)



Well, as we discussed in the last lesson. The algorithm steps for Gauss Siedel method are first, what we need to do is, to find the bus injections P i and Q i. P i and Q i from the known values of generation, that is P g i, Q g i and the nodes P d i and Q d i at bus i. So, once, we know these quantities, we can calculate P i and Q i, P i will be P g i minus P d i and Q g i minus Q d i will be Q i.

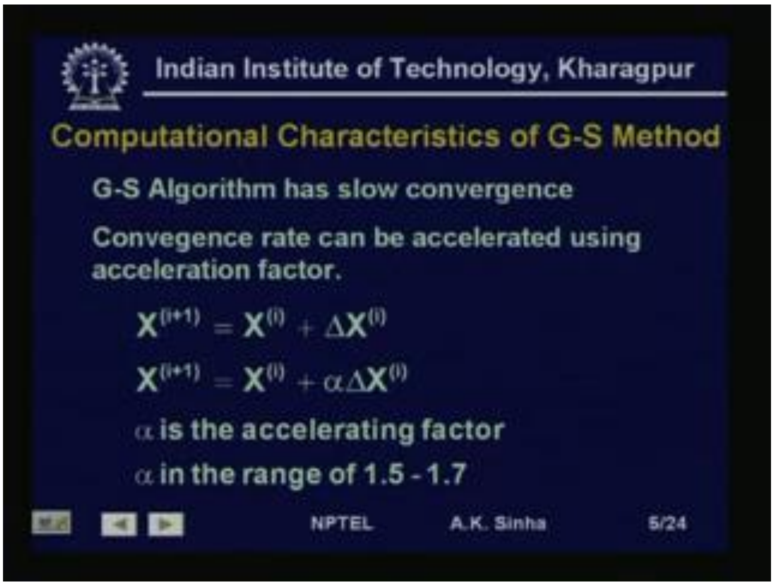
So, we first need to calculate a bus injections, after that, we need to form the Y bus matrix, for the power network. Then, we will set the initial voltages as we said in the last lesson. Normally, we use a flat voltage start, where the magnitude V i is set as one per unit and the angle delta i is set as 0 degrees. After we have set the initial voltage values, we iteratively solve the equation given here. V k at iteration i plus 1 is equal to 1 by Y k k into P k minus j Q k.

BY, V k conjugate at iteration i minus summation Y k n, V n at iteration i plus 1 from for n is equal to 1 to k minus 1. And plus summation Y k n into Vn i for n is equal to k plus 1 to N. Now, here, what we are seeing is, we are trying to use the latest available value. So, if we are solving for equation number, say 5, where k is 5, then V 5 if we are solving, then we already have solved for V 1, V 2, V 3, V 4.

So, in that case, what we have is for V 1, V 2, V 3, V 4 we will be putting the latest value. That is, i plus 1th iteration values, whereas for other voltages that is V 6 to Vn, we will be having only ith iteration values available. So, we will substitute those values and obtain new value, that is at i plus 1th iteration for V 5. So, in this way, for all the voltages, we need to solve this equation and obtain the new values, like this.

And we will continue this iterative solution, till V k i plus 1 minus V k i. That is the values for K plus 1th, sorry, i plus 1th iteration and ith iteration are very nearly equal. That is the difference between them is very, very small or less than a set value, which we have used as a tolerance level. So, this iterative solution is carried out, still we have a convergence obtained or the value change is less the tolerance value. That we have specified.

(Refer Slide Time: 06:30)



Now, we will talk about computational characteristics of Gauss Sidel mehtod. Gauss Siedel algorithm has one problem, that it is convergence is somewhat slow. In the sense that, suppose if you have a very large system. Suppose, you are having a system with more than 100 buses or a 100 bus system, let us take for example. Then, the number of itertaions, which one needs to solve these node flow equations for hunderd buses.

We will have 100 equations to be solved and normally, this will take the number of iterations to reach a reosanable convergence will be also of the same order that is about 100 iterations. So, as the system size, becomes larger and larger, the number of iterations also are keep on increasing for this kind of a method. And this is one of the reasons, why this method is not so much referred, nowadays for very large system analysis.

Of course, we can accelerate the convergence to some extent by using an acceleration fact. So, what we do is, that we have the updates X i plus 1 is equal to X i at delta X at iteration i. So, what we do is, instead of adding this delta X, that is what we have is, X i plus 1 minus X i. That is, whatever value we have got, say for, sorry, for V k at i plus 1 and this value for V k at i is known. So, V k at i plus 1 minus V k at i will give the value of delta V k, now, delta V k at iteration i.

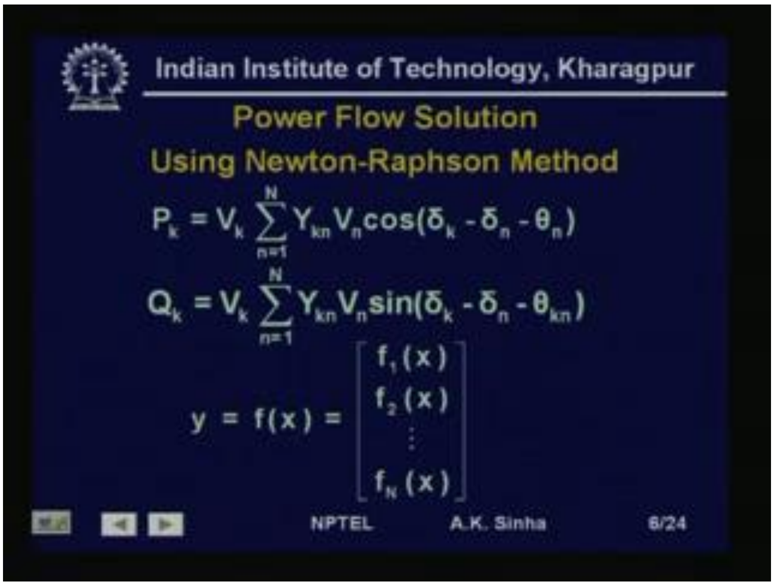
Now, this instead of using this value, what we do is, we add instead of delta X i or delta V i. We are adding alpha times delta V i to obtain this V at i plus 1. So, if we are doing that, then what we are doing is, we are trying to reach the solution faster, for accelerate the process of iteration.

Now, this has two aspects, sometimes this can lead to divergence, as well if you choose the value of alpha very large. Then, what can result is the divergence, that is you may not reach any convergence solution at all. Though, in many cases, if we alpha in the range of 1.5 to 1.7, it has been found that. Generally, these values will provide a much better convergence, than if we are using no accleration. That is alpha is equal to 1or any.

So, some accleration of Gauss Siedel algorithm, can be achieved by using an acceleration factor added to the updates that we get. However, there is still is not very good for large system analysis, because still the number of iterations required is much larger. And therefore, the time needed for the solution is much larger. And therefore, their foresearch for better methods for solving this float flow of power flow problem.

And one of the methods, which provides almost constant number of iterations for any size of system. That is the number of iteration is independent of the size almost independent of the size of the system. One of the method, which provides this kind of a property is the Newton Raphson, which we will talk now.

(Refer Slide Time: 11:12)

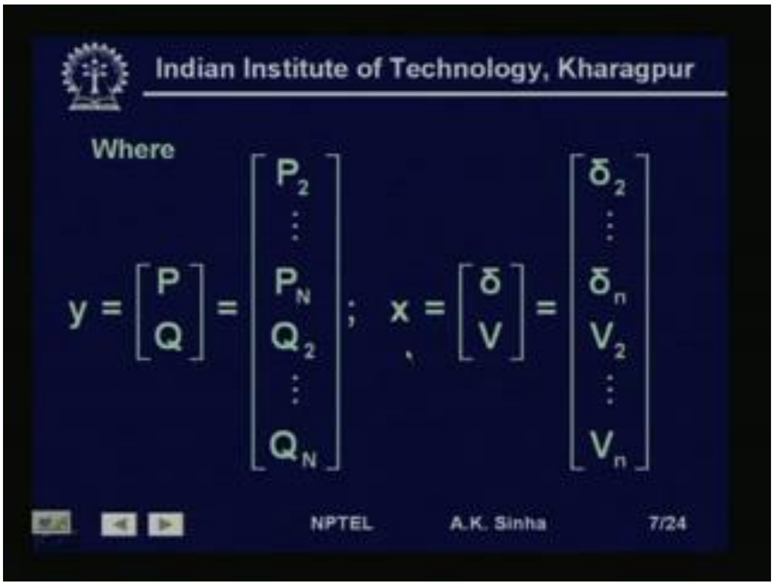


So, we will see, how we apply newton Raphson method to power flow analysis. Now, here, as we had seen, if we write the power flow equation for each bus in terms of power injections, real power injection and reactive power injection, seperately. Then, we get real power injection P k is equal to V k summation, Y k n, V n cos, delta k, minus delta n, minus theta k n, for n is equal to 1 to capital N.

Where, capital N is the number of buses and delta k and delta n are the voltage angle associated with the voltage at bus k and bus n, theta k n, I am sorry, this should be theta k n, theta k n is the angle of the admittance Y k n. Similarly, the reactive power injection at bus k will be given by V k summation Y k n, V n sin delta k minus delta n minus theta k n, for n is equal to 1 to capital N.

So, these two equations are basically providing us a relationship between power injection at any bus k and the voltage magnitudes and the network parameters. Now, we can write this equaiton as in the general form as y is equal to f x. Where, f x is a vector f 1, f 2 f three upto f N. That is for each bus, we have one funciton, which relates the real power and reactive power act that bus. So, we have n such functions.

(Refer Slide Time: 13:31)

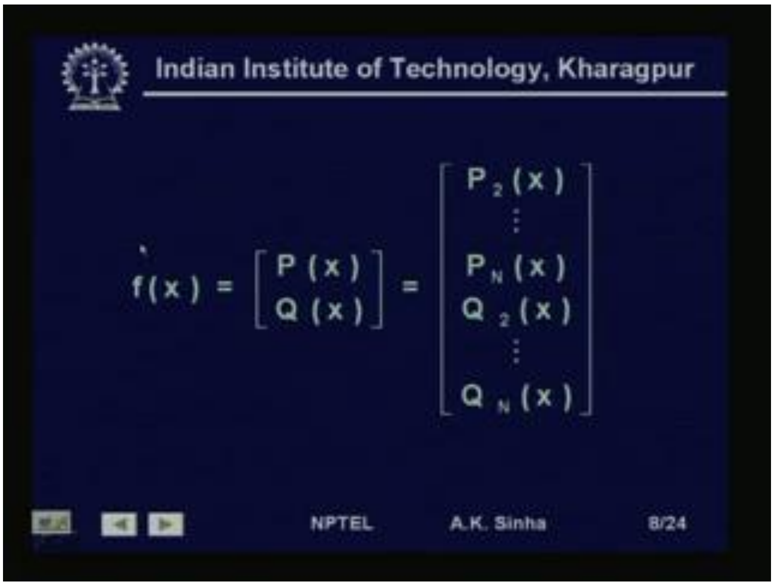


So, y is equal to f x is the general form of this equation. Now, where we have y equal to P and Q, vector P and Q are basically from P 2 to P N, P 2, P 3 upto P N and Q will be from Q 2 to Q N. That is Q 2, Q 3, Q 4 upto Q N. Now, here, what we have done is, we have assumed that bus 1, is a swing bus of slat bus. That is, for this bus, we have specified the voltage magnitude, as well x angle, the this is a reference bus.

So, it is angle is generally assumed to be 0 and the voltage magnitude is also specified for this bus. We have already discussed, why we need this swing bus in the system for power flow analysis. So, this is why is this a vector of P's, Q's and x is a vector of the voltage angle and the voltage magnitude, which again can be written as a vector from delta 2 upto delta n. That is delta 2, delta 3, delta 4 upto delta n and V 2 to V n. That is V

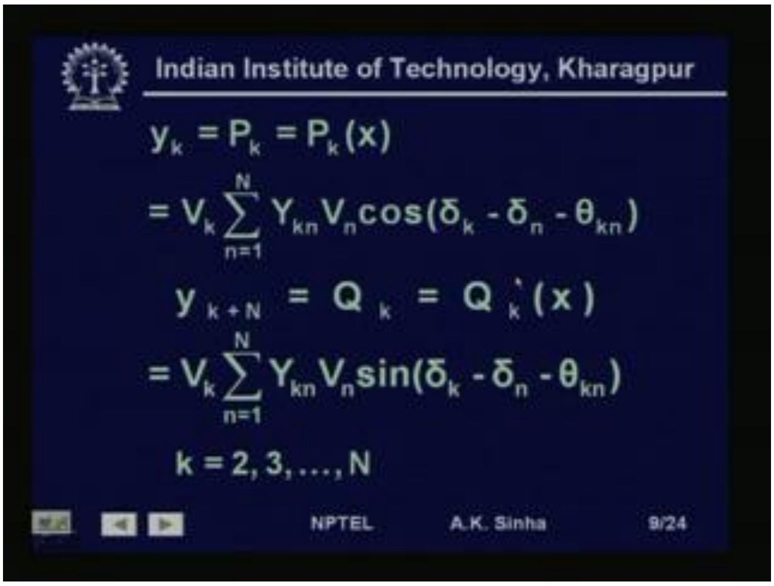
2, V 3, V 4 upto V n for all the n buses in the system. Since, delta 1 is 0 and V 1 is specified, which is normally 1 or maybe very near to 1, whatever value, it is that value is specified.

(Refer Slide Time: 15:18)



Therfore, we can write f x is equal to to P x, Q x. where, the P x, Q x of this vector f x is P 2 a function of x upto P N. All these are function of x, Q 2 to Q N; all these are also funciton of x, where x is basically the voltage. Angle delta and the voltage magnitude V, at all the buses.

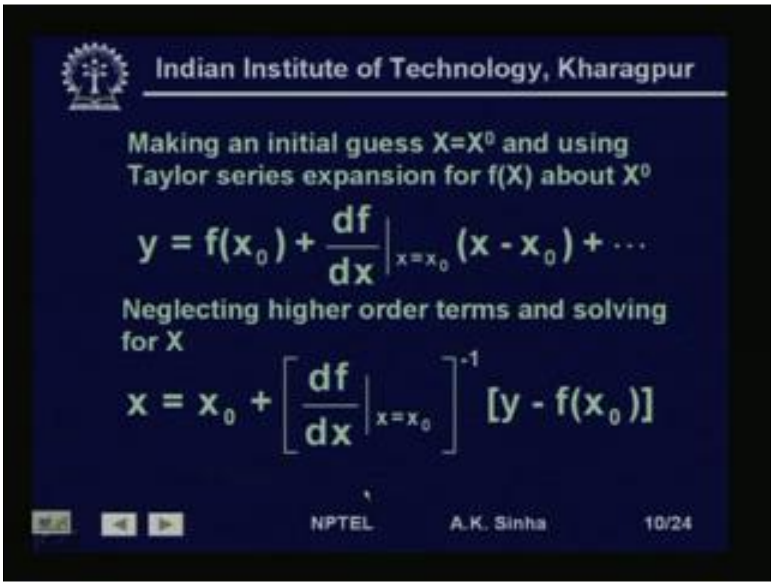
(Refer Slide Time: 15:50)



So, now we can write y k, that is the element, k th element of this y matrix will be y k is equal to P k, which is a function of x. So, we can write it as P k in bracket this X, where this P k is a function of X. And this is equal to V k summation Y k n, V n cos delta k minus delta n minus theta k n. This is already, we had explained from summation is from n is equal to 1 to capital N.

Now, y k plus N is Q k. That is we have, since we had arranged our y in such a way. That from 2 to N, we have P and then after that N plus 2 to 2 N, we have the Q values. We have no values for P 1 and no value for Q 1, because this is one bus, 1 is a slat bus. So, we have y k plus n equal to Q k is equal to Q k a function of X, which we can as we have seen earlier, can be written as V k, summation Y k n, V n sin delta k minus delta n minus theta k n. For k varying from 2 to N, that is for the all the buses, except the slat bus, which is bus 1 in our case.

(Refer Slide Time: 17:43)



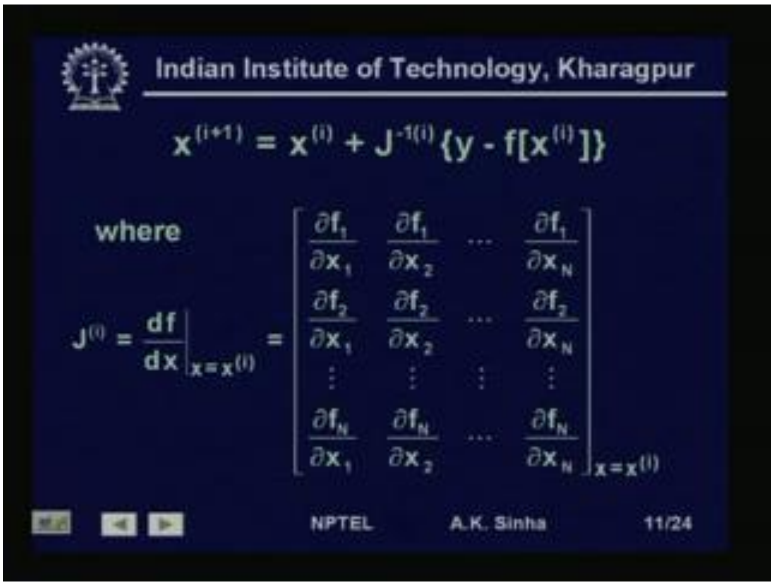
Now, if we make an initial guess X is equal to X 0. And then use the Taylor series expansion for f X about this initial guess of the state X. Then, we can write y is equal to f X 0 plus d f by d x at x is equal to x 0 into X minus X 0 plus higher order terms will be there. That is, d 2 f by d x 2, factorial 1 by 2, X minus X 0 square, all those terms higher order terms will be there.

Now, if we neglect the higher order terms. That is, if we are assuming that our X 0 is close to the true value X, then X minus X 0 square and X minus X 0 cube all those terms are going to be much smaller and therefore, they can be neglected. So, neglecting higher order terms and solving for X. We get the solution X is equal to X 0 plus d f by d x at X is equal to X 0 inverse into y minus f X 0. That is from this expression itself.

If we forget these terms, then we can write this X on this side and we can take this X 0 on this side. So, we have X 0 and this y minus f X 0, we can write here. So, y minus f X 0 will be this side and then d f by d x. That X is equal to X 0 into X minus X 0 is there. So, then again taking this X 0 on this side, so we will have X is equal to X 0 plus d f by d x inverse into y minus f X 0.

So, what we are doing is, taking this f X 0 on this side, so y minus f X 0 and then we are pre multiplying it by the inverse of this. So, we get d f by d x inverse into y minus f X 0. That will be equal to X minus X 0. Therefore, X is equal to X 0 plus this term. So, this is what we get as the solution for X. Now, since our guess may not be very close. Therefore, what we do is try to solve this using an iterative method.

(Refer Slide Time: 20:37)



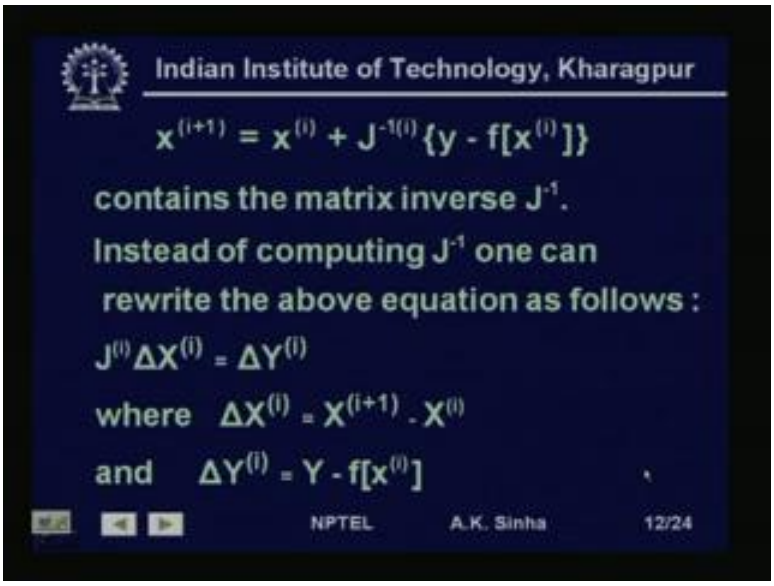
So, the iterative solution, this for iterative solution this equation, becomes likes this. X for i plus 1th iteration is equal to the value of X for i plus 1th iteration is equal to the value of X at ith iteration plus J inverse. Where we are writing d f by d x inverse as J inverse, so d f by d x is our J. So, this is J inverse at ith iteration into Y minus f X at ith iteration.

So, this is the iterative solution. That we will get for at each step, that is, at each iteration, we will update X and we will keep doing it till we reach a convergence. That is X i plus 1 minus X i becomes very small or becomes less than our specified tolerance. Now, here as I said earlier, this J is computed or J i is basically a value of the first order derivative d f by d x, for X is equal to X at ith iteration.

So, substituting the value of X at ith iteration, in the equation, I am taking, it is derivative, we will get the J i. That it so J i can be defined as del f by del X I, del f 1 by del, sorry del f 1 by del X 1, del f 1 by del X 2. That is, for all the N equations that we have, we will have these partial derivatives of f with respect to X and that is X 1 up to X N. So, we will have del f 1 by del X 1, del f 1 by del X 2 and so on, like del f 1 by del X N.

Similarly, for the second set of equation, that is del f 2 by del X 1, del f 2 by del X 2, del f 2 by del X N and so on. Till the nth equation we have is, del f N by del X 1, del f N by del X 2 and so on, till del f N by del X N. This all we are calculating by substituting the value of X at ith iteration, that is X is equal to X i.

(Refer Slide Time: 23:11)



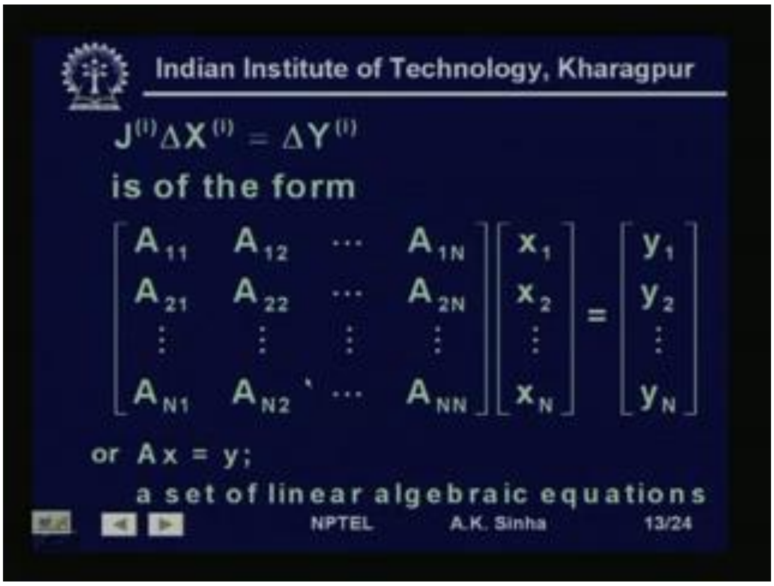
Now, this equation, as we have seen here, X the updated value or the next iteration value X i plus 1 is equal to X at ith iteration plus J inverse at ith iteration into Y minus f X i. Now, here we have got a J inverse term and as we have seen J is a n by n matrix. In our case, since we are having n buses in a system, if we have n number of buses in a system, then we have how many equations. We have except for the slack bus; we have two equations for each bus. So, that is, if you are assuming that all the buses are PQ buses.

So, total number of equations that we have will be equal to 2 into n minus 1. So, that is for a 100 bus system, we will have 2 into n minus 1. That is 99, that is 198 equations. So, we will have a 198 by 198 matrix, representing this J. Now, this matrix is a large matrix, for a large system this will be even much larger matrix. So, as we here, we need an inverse and finding out an inverse for a very larger matrix is very time consuming process.

So, in power system applications, we generally do not find the inverse of the matrix, rather we try to solve it by using some other means; we will see how we do that. So, this equation contains the matrix inverse, J inverse. Instead of computing J inverse, one can rewrite the above equation as follows, that is this equation can be rewritten. Now, that is, if we write X i plus 1 minus X i as delta X. Then, if we take this on this side, then we have J i into delta X i is equal to y minus, f X i is nothing but, delta y.

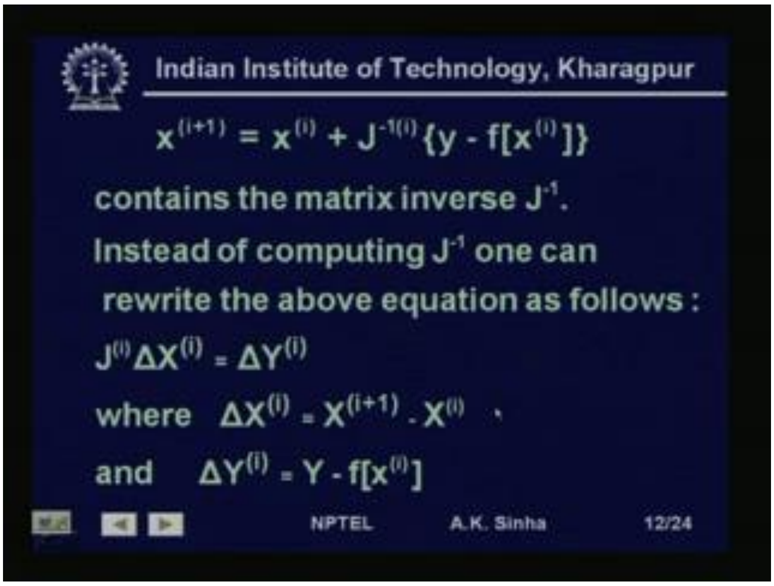
So, is equal to delta y, where we are writing delta X i plus 1 minus X i and delta y i is minus f X at i or ith iteration. So, if what we are seeing here is, now we have a set of equations, which can be written in this form, where we do not have the inverse of J, but we have J matrix. So, we need to solve for delta X, using this equation.

(Refer Slide Time: 26:00)



Now, if you look at this equation, that is J i delta X i is equal to delta Y i is of the form A x is equal to y, where A is just matrix J. So, instead of J, we are writing here as a general form, this is of the form A, which is a N by N matrix, delta X here is the X values is equal to y, where these are the delta y values which will be there. So, at each iteration what we need to do is, solve a set of equations A X is equal to y, these are a set of linear algebraic equations.

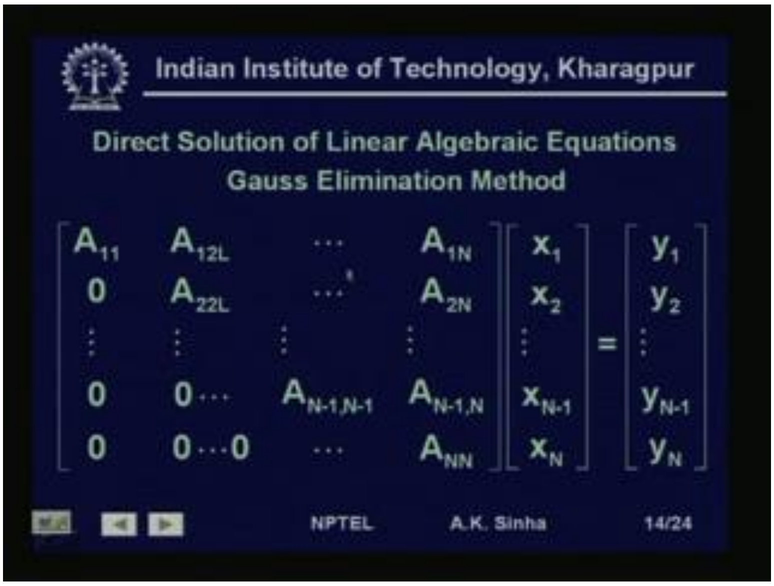
(Refer Slide Time: 27:11)



So, basically, if we look at the Newton Raphson iterative scheme, then what we see is, this is the solution that we need to obtain. This X i plus 1 is, what we need to get, we can get delta X i by solving this sort of an equation or a set of linear algebraic equation. And once, we know this delta X, knowing X at ith iteration and knowing delta X at ith iteration, we can get X i plus 1 is equal to X i plus delta X i. Therefore, we can always get the updates in each iteration.

So, if you see the Newton Raphsons solution method, finally was down to solving a set of linear algebraic equation. That is what we have done is a set of non-linear algebraic equations are being solved as a set of linear algebraic equations, where we are linearizing the equation at each and every step of iteration.

(Refer Slide Time: 28:24)



Now, how do we solve this set of equations? That is this A x is equal to y, how do we solve this kind of a system of equations. Now, what we need to do this is, if we can one method as we said earlier is take the inverse of A pre multiply it on both sides. Then, X is equal to A inverse y, that is the easiest way to do it. But, finding inverse is computationally much more complex and time consuming.

In fact, finding inverse for a n by n matrix, the time required is proportional to N cube. So, if the size of the matrix increases, then computation time for finding inverse also increases much more. Therefore, instead of finding inverse in that way, we need to find some other way of solving it. And one of the ways as we said, here is direct solution of linear algebraic equation, using Gauss elimination method. So, here we are trying to see, how we can use Gauss elimination to solve this set of equation.

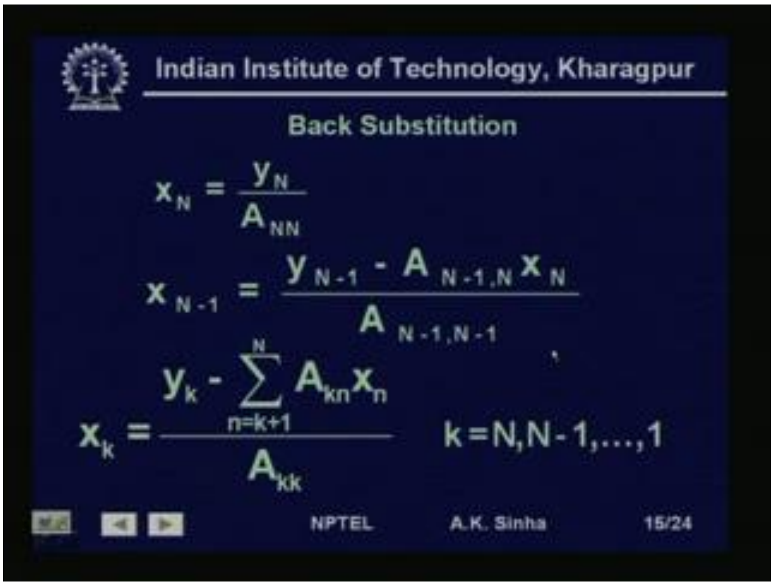
Now, what this Gauss elimination tries to do is, tries to triangularize this matrix A, this matrix A, which is a n by n matrix into an upper triangular matrix. That is, all the elements, below this diagonal A 11, A 22 A N N are made 0. So, all the elements here, A 22, A 21 up to A N 1, then A 23, sorry A 32 up to A N 2 and so on, all these will be made 0. If we can do that, then we get an upper triangle matrix as shown here, where all the elements, below the diagonal are 0.

Now, if we can do this kind of a thing, then finding the solution for this set of equation is very easy. Because, you can see we know this Y N, if we want to find X N, then we have from the last equation is A N N into X N is equal to Y N. So, X N is equal to Y N by A

N N. And once, we know this value of X N and then we can substitute this value into this location.

Because, here we have A N minus 1 into N minus 1 into X N minus 1 plus A N minus 1 into N into A X into X N is equal to Y N minus 1. So, knowing this value, substituting it here, we have only 1 unknown X N minus 1, which can be again found out and so on, we can proceed backward from the nth element. And finally, we can go up to the first element and therefore, we can find out all the unknowns X 1 to X N.

(Refer Slide Time: 31:43)

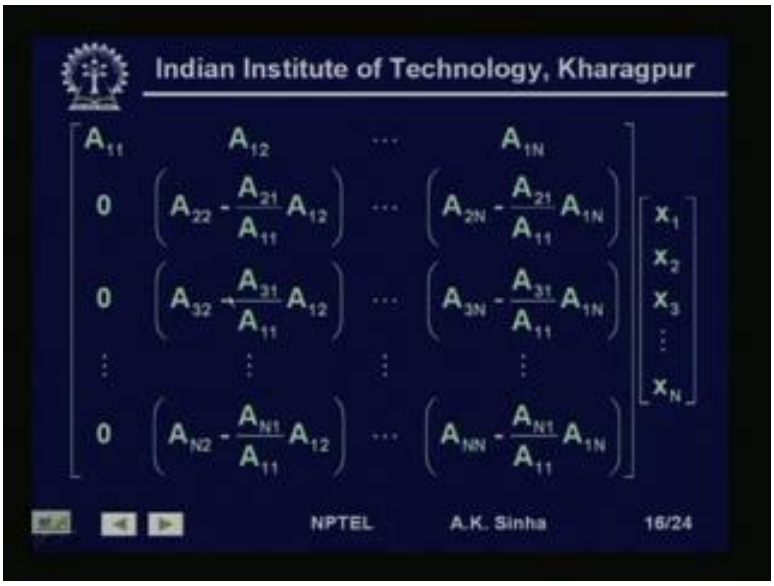


So, this is, what we call the back substitution method. So, here as I said earlier X n will be given by Y n divided by A N N. X N minus 1 will be given by Y N minus 1 minus A N minus 1 into N into X N. Now, X N value is already known from here divided by A N minus 1 into N minus 1. Similar way, we can write for any general element X k. This will be equal to Y k minus summation A k n into X n for N is equal to K plus 1 to N divided by A K K for K is equal to N N minus 1, so on up to 1.

That is, we proceed backward from N th element to the first element. That is starting with the last equation; we go backward up to the first equation. In this way, we can find out all the unknowns X. Now, how do we triangularize this matrix? That is the problem that we have now. So, in Gauss elimination, what we try to do is, since we have this, first we have this, A here like this. Now, what we want to do is, first we will try to make all these elements below A 11 as 0.

Now, for doing this, what we can do is, we can multiply this row by A 21 and then subtract from this.

(Refer Slide Time: 33:37)

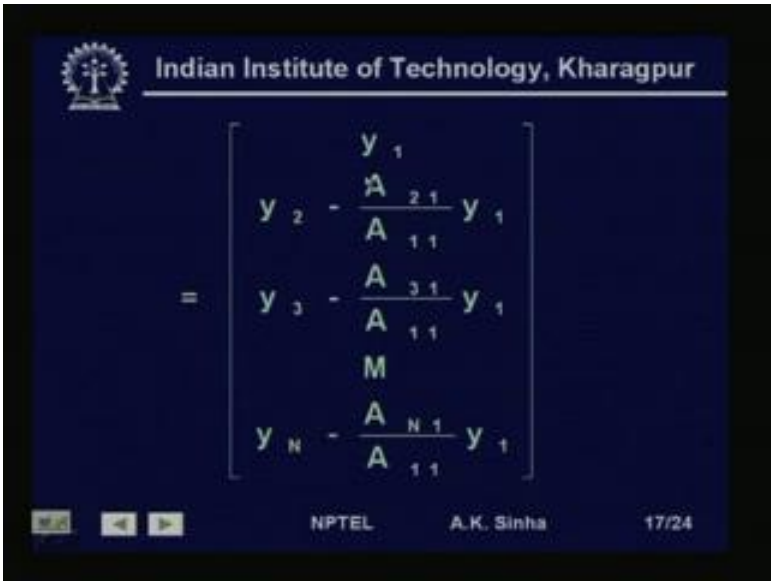


We will see, how we do that? That is, we can multiply this by row by A 21 by A 11. So, we multiply this row by A 21 by A 11 and then subtract from the second row. Then, we will get this as A 21 minus A 21, so this becomes 0. This element will be A 22 minus A 21 by A 11, which is this all the elements of the this row, has been multiplied by that into A 12 will be the element of this row.

So, A 22 minus this element will be the new element at this position and so on. For A 2 N minus A 21 by A 11 into A 1 N will be this element, when we multiply this row by A 21 by A 11. So, when we subtract it, this will be the new value of the element A 2 N. Similarly, for the third row, we multiply this by A 31 by A 11 and then subtract from the third row elements.

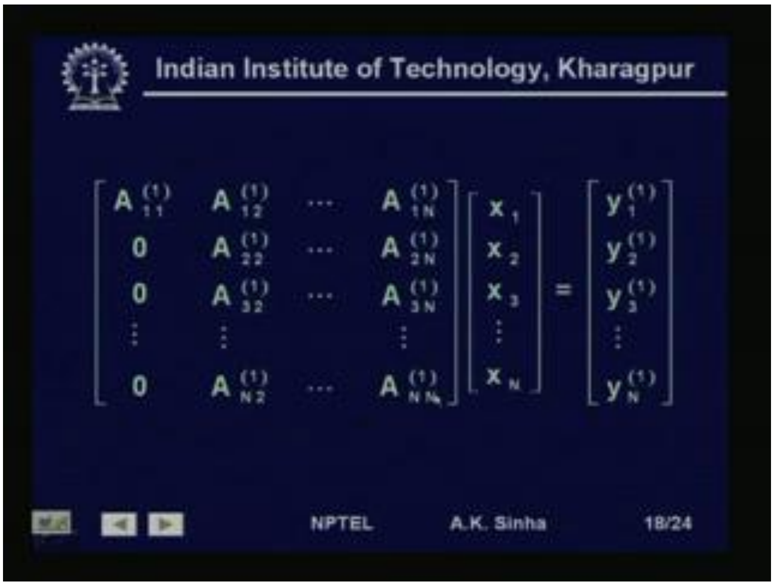
So, all the elements of row first, row 1 are multiplied by A 31 by A 11 and then it is subtracted from the third row and so on. We can do up to Nth row, then what we will get is 0's in the first column below A 11. So, all the rows here, we can eliminate the first column elements, like this.

(Refer Slide Time: 35:28)



So, we can write this, when we are doing this. Of course, we are also multiplying and doing the same subtraction for the y also. So, we have y 2 is equal to A 21 by A 11 into y 1, y 3 will be equal to y 3 minus A 31 into A 11 into y 1 and so on up to y N minus A N 1 by A 11 into y 1. So, the column vector y of the known variables is also getting modified in the same way.

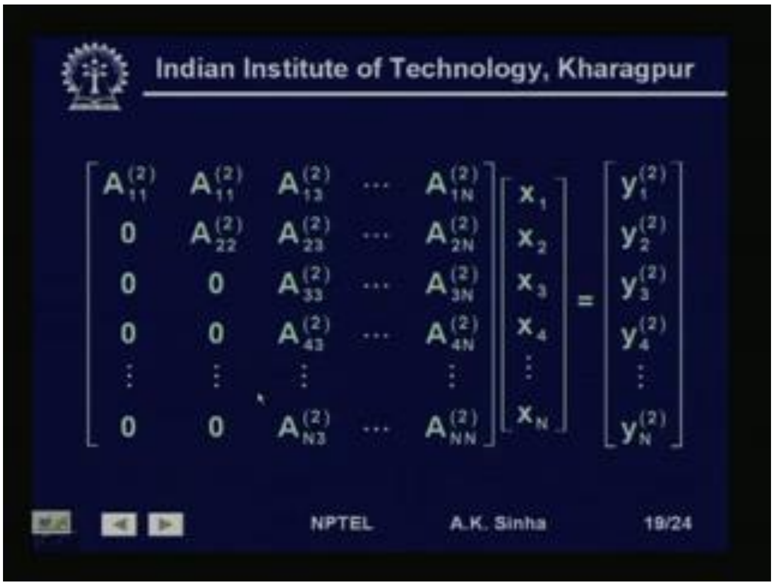
(Refer Slide Time: 36:10)



So, now, what we have, after we have done this process, sorry, after we have done this process, then we can rewrite these elements as A 22, 1 A 2 N, 1. That is the modified value, after the first step of elimination. That is the first step of triangularization, we get all these elements modified elements, that we can write as with a superscript one here. So, this is now the set of equations, that we have, where these y's are also modified values.

Now, what we need to do now is, below A 22, all the elements in the second column have to be made 0. So, in the same fashion, we can multiply the second row by A 32 by A 22, and then subtract from the third row and so on. That is for fourth row, A 42 by A 22, we will multiply this row and then subtract from the fourth row and so on.

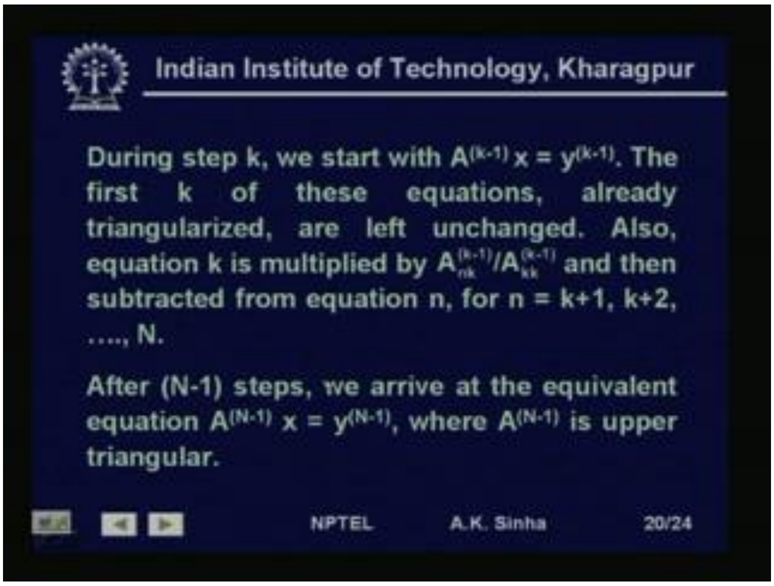
(Refer Slide Time: 37:33)



So, if we do that, then what we will get is, all the elements in column 2 will become 0 and in the same way, we have to modify the column vector y also. So, we will get a new set of y's. Of course, y 1 will remain same y 2 will remain, whatever as y 21, but we are writing these new values. Because, all the rows, which are already below which the diagonal of which we have already eliminated all the elements are not to be processed again.

So, this row is not getting processed, only this row, we had processed using this row, all the rows below this or all the equation below this and so on. If we can do that, then finally, we will get a triangular matrix.

(Refer Slide Time: 38:27)



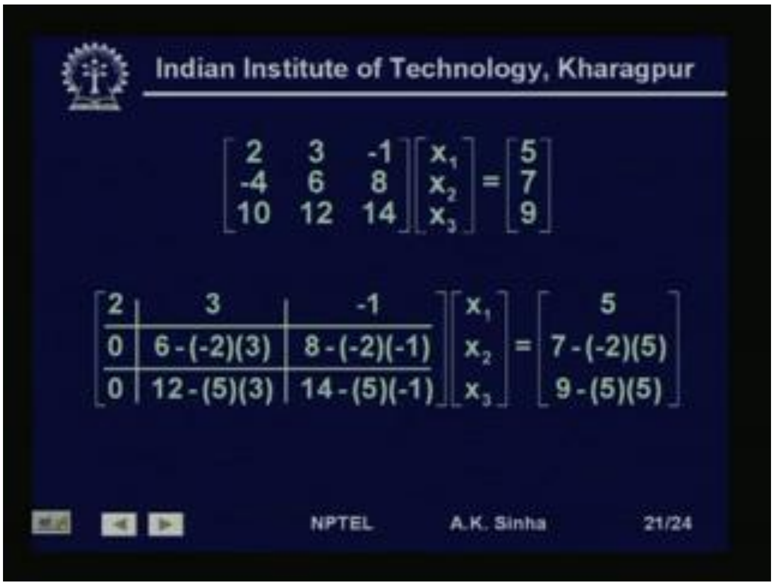
So, we can write this process as such during step k, we start with A k minus 1 X is equal to y k minus 1. This is k minus 1 is the already affected elements, that we have done at k minus 1th step. So, A k minus 1 X is equal to y k minus 1. The first k of these equations already triangularized is left unchanged, as we have seen here, if we have already triangularized this row.

So, when we are taking the second equation, we do not do anything to the first equation. When, we are taking the third row third row equation and then first and second, which have already been triangularized will be left unchanged. So, then you see here, this is not changed, only below this all are getting changed. So, are left unchanged. Also, equation k is multiplied by A n k at for the k minus 1th value, whatever is available divided by A k k at k minus 1th value.

And then subtracted from equation n, for n is equal to k plus 1, k plus 2 up to capital N. So, this is the process, that we follow, that is we take equations 1 by 1 and we eliminate columns for them. That is the all the elements in the column below the diagonals are reduced to 0. So, this is, how we process it. So, after n minus 1 steps, we arrive at the equivalent equation A N minus 1 X is equal to y N minus 1, where A N minus 1 is the upper triangular matrix.

So, this is the way, we work with the Gauss elimination method. So, by using this Gauss elimination method, what we are doing is, we are triangularizing the matrix and obtaining an upper triangular matrix. Once, we have this upper triangular matrix, then we can find out the value for the X or the unknowns, which we need to find out in A X is equal to to y.

(Refer Slide Time: 41:21)



So, let us take a small example. Here, we have a 3 by 3 matrix, our A matrix is a 3 by 3 matrix. The first row is having 2, 3 minus 1, that is the equation, that we have is 2 X 1 plus 3 X 2 minus X 3 is equal to 5. The second equation we have is, minus 4 X 1 plus 6 X 2 plus 8 X 3 is equal to 7. The third equation is 10 X 1 plus 12 X 2 plus 14 X 3 is equal to 9.

So, in matrix form we write this. This is our A matrix, this is the unknowns the X 1, X 2 X 3, that we need to find out and this is the y vector, which for which the values are known. Now, what we do we need to triangularize this matrix? That is these elements these three elements need to be eliminated. So, what we do, first we take this matrix.

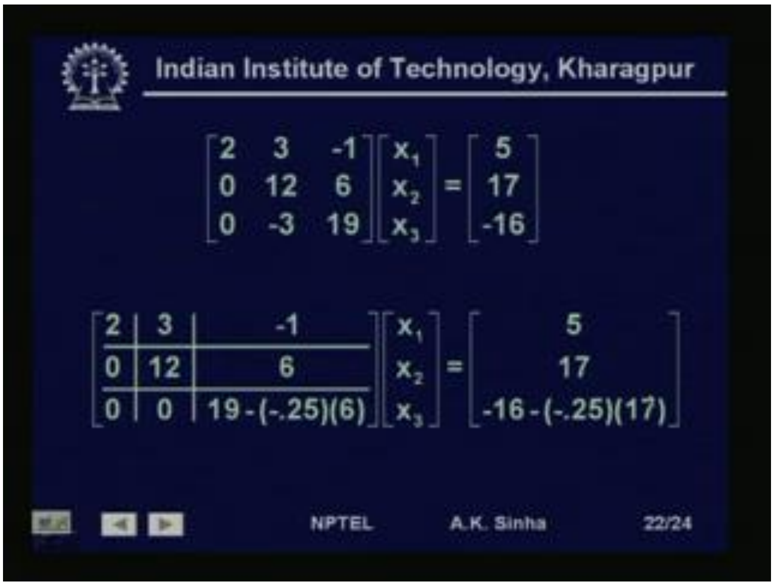
Now, first row is like this, we do not need to do anything to the first row; we need to eliminate this or make this 0. So, what we are going to do is, we are going to multiply this row by A 21 divided by A 11. So, A 21 divided by A 11 will be equal to minus 4 divided by 2. So, that is equal to minus 2. So, we are multiplying this equation by minus 2 and subtracting it from this. So, when this, we minus 2 into 2 is minus 4. So, when we subtract it from minus 4 we get 0.

Then, here we have 6 minus 2 multiplied by 3. So, 6 minus, minus 2 multiplied by 3 and here, we will have 8 minus, minus 2 multiplied by this minus 1. So, this is the value that we will get. Then, again for eliminating this, we have to multiply this by 10 by 2. That is

5. So, multiply this equation by 5, then we get 10, here we get 15, here we get minus 5. So, when we subtract it from this, we get 0 here, then 12 minus, this is 15, that is 5 into minus 5 into 3 and this is 1 into 5, so that is minus 1 into 5. So, here we have 14 minus 5 into minus 1. So, this is what we get.

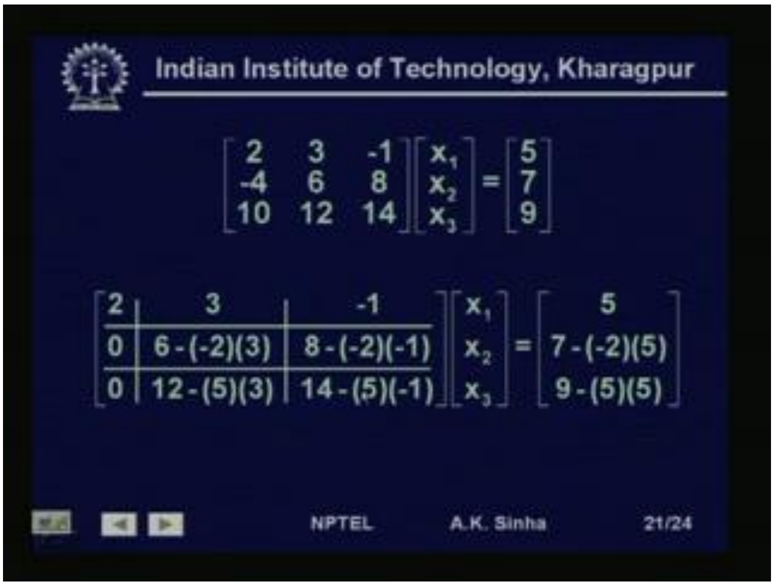
Now, when we are doing this equation, we also have to work on this. So, here, we had multiplied this equation by that is we had multiplied this equation by minus 2 and then subtracted from this. So, 7 minus, minus 2 into 5, so minus 2 into 5, this equation is multiplied by minus 2. So, this becomes minus 2 into 5. That is subtracted from 7. Similarly, when we are eliminating this, we had multiplied this equation by 5. So, what we have is 9 minus 5 into 5, when we multiply this equation by 5, so 5 into 5, so 9 minus 5 into 5.

(Refer Slide Time: 44:51)



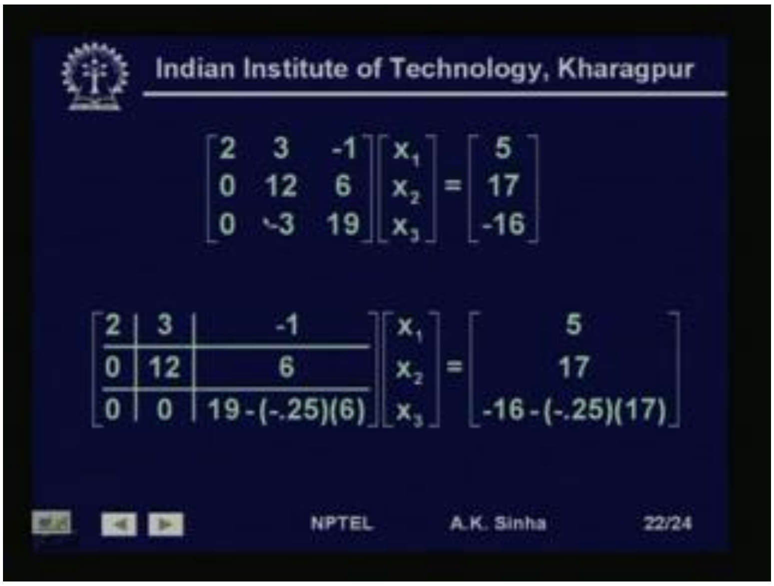
So, this if we do, we get this set of equation now as 2, 3 minus 1, this row does not get changed. These two values have become 0. This is from here 6 minus, minus 6, that is 6 plus 6, 12, 8 plus 2. So, this is plus 2. So, 8 minus 2, that is 6. So, what we have sorry.

(Refer Slide Time: 45:16)



What we have here is, that is 12 and this 12 is 8 minus 2, 6. So, this is 6, then this term is 0, this term is 12 minus 15. So, that is minus 3 and this term is 14 plus 5, that is 19. So, we get minus 3 and 19. Similarly, 7 minus of minus 10, that makes it 17, 9 minus 10, that makes it 9 minus 25, that makes it 60.

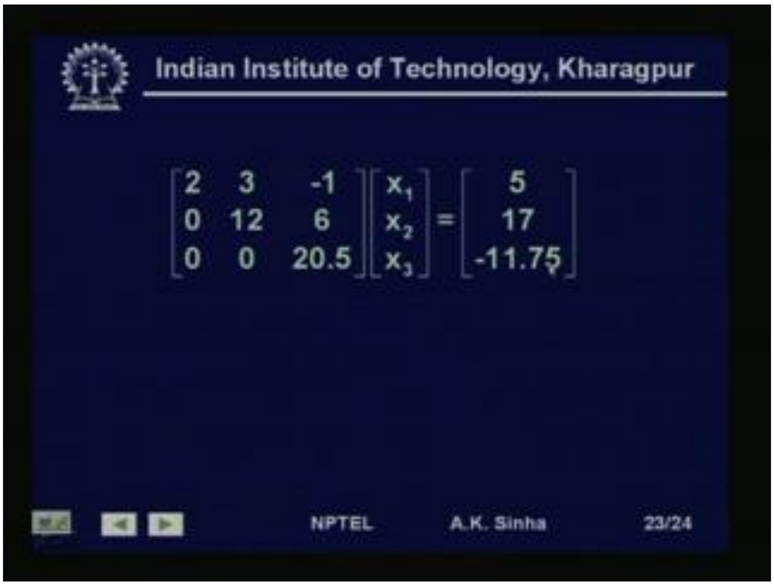
(Refer Slide Time: 45:53)



So, we have minus 16 here, 5 and 17 and minus 16. So, now this is the set of equation that we have now we need to do is, eliminate this element. So, that we get a triangular matrix, upper triangular matrix. So, for eliminating this element, what we do is, we multiply this equation by minus 3 by 12 and subtract from this. So, minus 3 by 12 is minus 0.25 or minus 1 by 4.

So, we multiply this equation by minus 0.25 and subtract from this, then this becomes 0 and we have this 19 minus, minus 0.25 into 6. So, this is what we have got here and here, what we have is minus 0.25 into 17 have to be subtracted from this minus 16, so minus 16 minus 0.25 into17.

(Refer Slide Time: 47:22)



When we do this, we get finally our set of equations as 2, 3 minus 1, 0, 12, 6, 0, 0, 10.5. Now, you see this is a triangularize matrix and the Y vector is also modified as 5, 17 minus 11.75. Now, once we have this set of equation, now what we need to do is, find out X 1, X 2, X 3. So, we will use the back substitution method. So, if you see here we have 20.5, X 3 is equal to minus 11.5. So, X 3 is equal 2 minus 11.75 divide by 20.5.

So, once we know the value of X 3, then from this equation, we have 12 X 2 plus 6 X 3. So, 6 X 3, we can substitute the values here is equal to 17. So, that can be brought on this side then 12 X 2, we have. So, divide this side by 12, you will get the value of X 2, once we know X 3 and X 2, we can find out the value of X 1. Because, this equation is 2 X 1 plus 3 X 2 minus X 3 is equal to 5. So, we can, since we know X 2 and X 3, now only unknown here is X 1. So, we can find that from that equation.

So, this is, how we solve a linear set of equations. And this is, Gauss elimination method is 1, which is preferred. There are other methods of solving this linear set of equations, which we call methods of decomposition. That is, we factorize this matrix a into lower and upper triangular matrices. That is LU factorization, sometimes we use LDU factorizations. There are other methods, which are called bifactorization method.

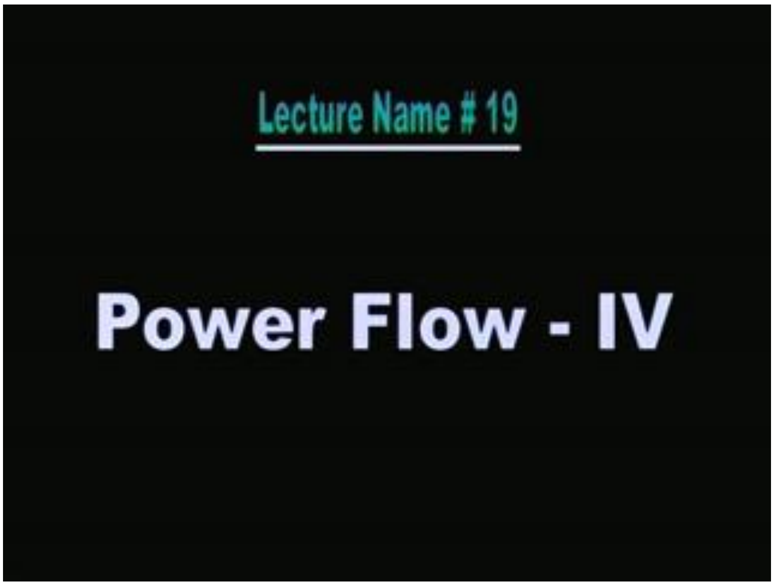
All these methods basic idea is, to avoid using inverse and try to find out the solution for a x is equal to y, using a direct solution method. That is by trying to factorize a matrix. So, this is of course, Gauss elimination is the simplest of these methods and is quite effective and is very much used for solving linear set of equations, which come out from the Newton Raphson load flow equations.

So, this is what we will do today and in the next class, will talk how to solve the Newton Raphson power flow equation, using this directs solution method. So, that is all for today.

Thank you very much.

(Refer Slide Time: 50:50)

(Refer Slide Time: 50:54)

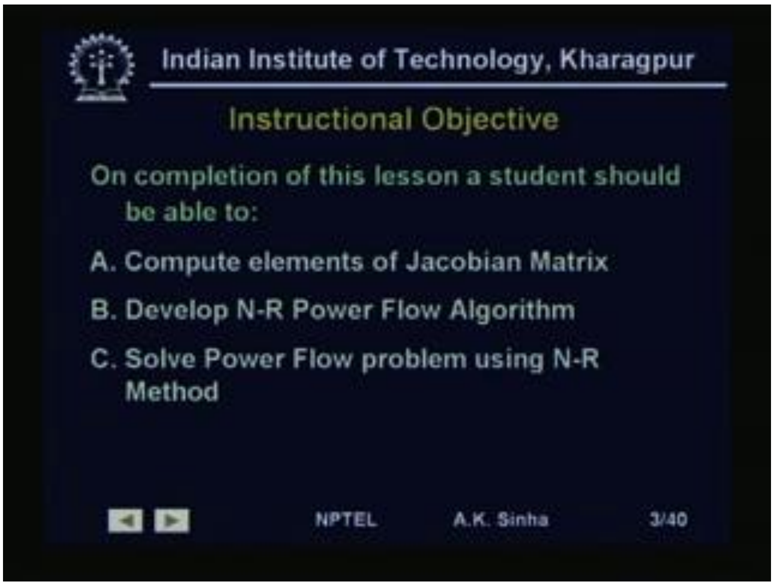


Welcome to lesson 19 on power system analysis. In this lesson, we will continue with power flow analysis.

(Refer Slide Time: 51:15)

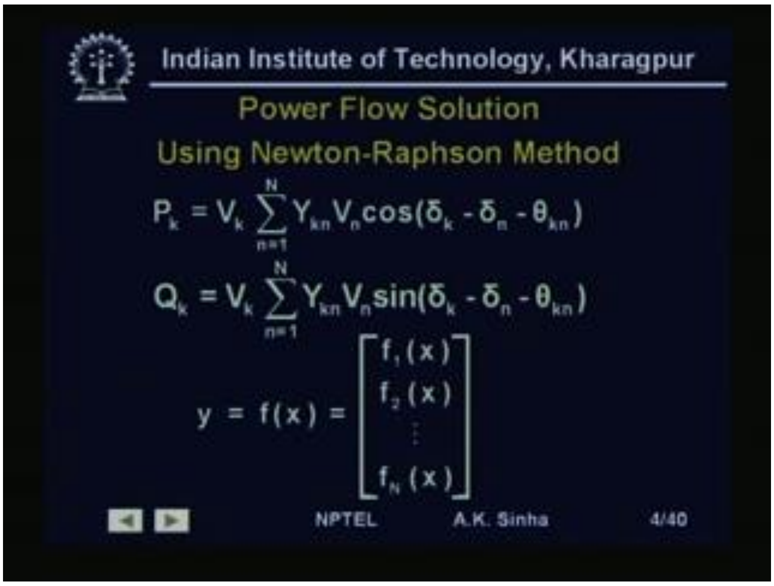
In this lesson, we will talk about a computation of Jacobian matrix for Newton Raphson method. And we will talk about the computational algorithm for Newton Raphson power flow. And we will work out one example, for a small power system, how we do an Newton Raphson load flow or power flow.

(Refer Slide Time: 51:35)



Well, once you complete this lesson, you will be able to compute the elements of the Jacobian matrix. You will be able to develop the Newton Raphson power flow algorithm and solve power flow problems using Newton Raphson method. Before, we go into the computation of Jacobian matrix elements. We recapitulate some of the equations. That we developed for the Newton Raphson load flow or power flow.

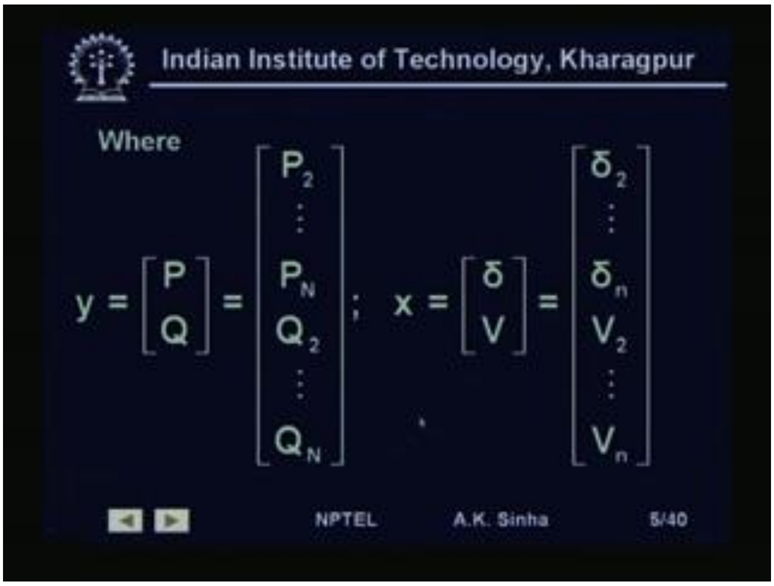
(Refer Slide Time: 52:17)



Well, as we had seen in the last lesson, the power flow equations. In terms of real and reactive power injections at any bus k is given by these relationships. So, which is a function of voltage magnitude at all the buses and the voltage angle, at all the buses as well as the admittance magnitude an angle for the transmission network.

Same is true for the reactive power, the equations can be written as shown here. We can write this as a function x as y is equal to f x, where for each bus, we will be writing these two equations. So, if there are number of buses, then we will write all these equations as f 1 x up to f N x, total number of equations required.

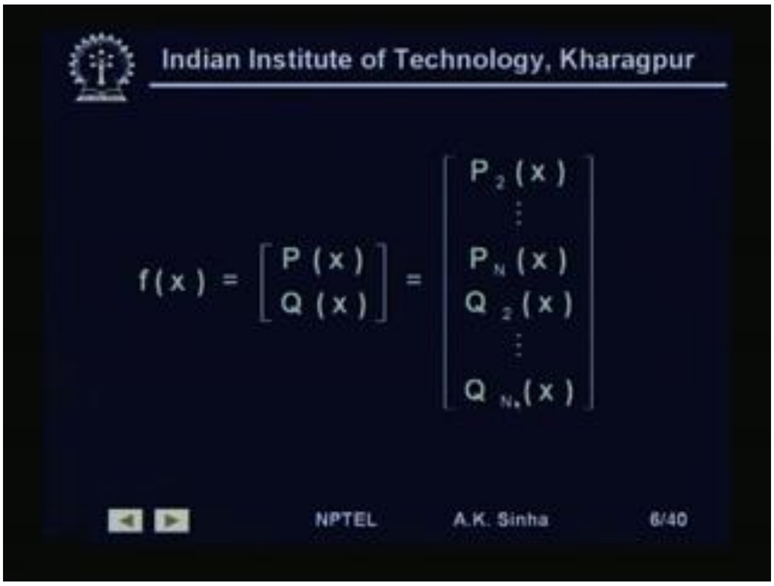
(Refer Slide Time: 53:28)



Where, y is basically a vector of real power injections P and reactive power injections Q. Here, we have assumed that bus 1 is a slack bus for our purpose. So, for slack bus since we know the voltage magnitude an angle, we do not have any real power or reactive power specified. So, we do not have any P 1 and Q 1 in this vector of y. So, we have P 2 to P N and Q 2 to Q N.

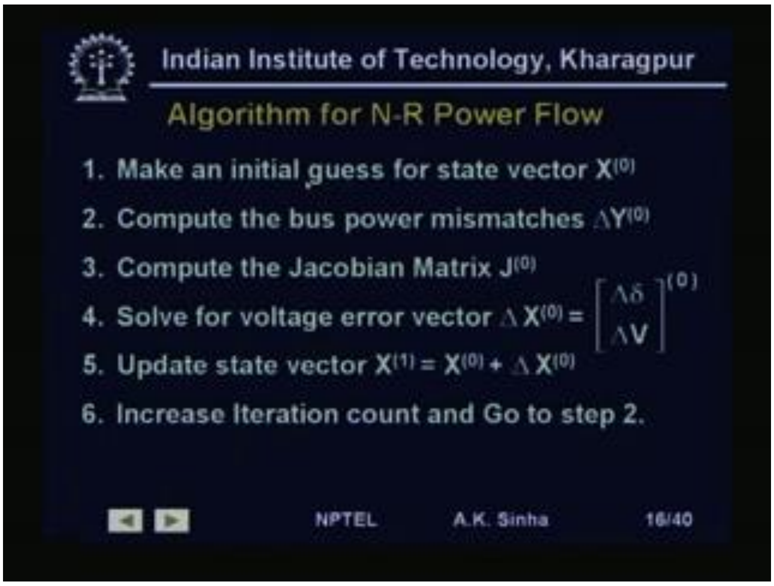
The unknown vector or the state vector is given by the voltage phase angle delta and V. Again, this delta is for all the buses except the slack bus for which we already know the angle, as we said earlier, we since the slack bus can be taken as a reference. So, delta 1 is normally taken as 0 degrees and so we have from delta 2 to delta N. The voltage angles the V 2 to V N, the voltage magnitude for all the buses, except the slack bus.

(Refer Slide Time: 54:49)



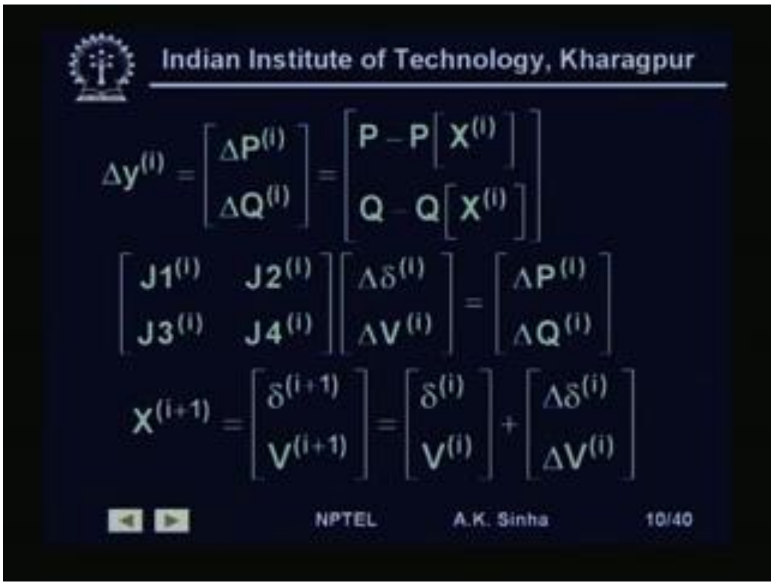
This can be written as f x is equal to P x, Q x in terms of P 2, which is a function of x up to P N and Q 2, which is a function of x Q, which is a function of x from Q 2 to Q N.

(Refer Slide Time: 55:05)



So, using these, we have made the initial guess for the state vector. Now, compute the bus power mismatches delta Y 0. Once we know the value, we have made the guess of V and delta, substituting the value of V and delta in to the equation for real power and reactive power.

(Refer Slide Time: 55:36)



We can find out the mismatch using this relationship P specified minus P calculated with the value of x known at the initial guess, Q specified minus Q calculated, with x at the specified value. So, using this, we can find out this delta P and delta Q, which is basically our delta y. So, compute the bus power mismatches delta y, next step is compute the Jacobian matrix J 0. That is with the initial guess state vector x.

That knows, all the voltage magnitude and angles with the guess initial value is known. We compute the value for the elements of the Jacobian matrix at the initial operating point. Now, solve for voltage error vector delta X, which is delta and delta V from the initial guess. That is the error vector, based on the initial guess, we can find out using as we have seen earlier, this relationship using this relationship, we solve for this.

Now, this is already known, but in general, the Jacobian matrix will be a sparse matrix, just like the matrix, which is which is a sparse matrix. Because, we know all the buses are not connected to all the buses in a large system. In fact, each bus may be connected to just a few other buses. And therefore, Y bus will contain large number of 0's. Similarly, the Jacobian will also contain large number of 0's.

These matrices as we said earlier are called sparse matrices and we take advantage of this sparsity of these matrices in solution to make the solution much more, faster. So, with this, we have completed the Newton Raphson load flow. Now, in the next lesson, we will talk about another version of load flow, which is very popular, which we call Fast Decouple load flow. So, with this we end today's lesson.

Thank you very much.