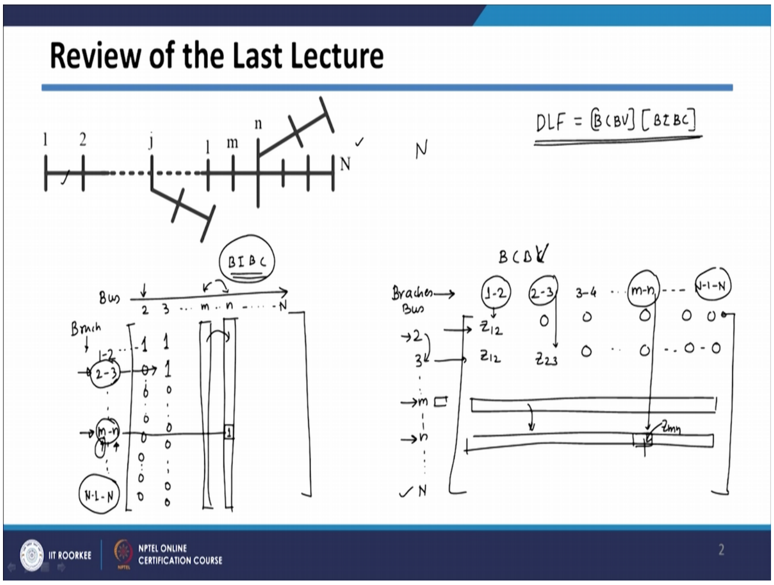
Electrical Distribution System Analysis
Dr. Ganesh Kumbhar
Department of Electrical Engineering
Indian Institute of Technology, Roorkee

Lecture – 31
Direct Approach Based Flow Analysis
Part III

Dear students, we are studying load flow method based on direct approach. And we have seen that in direct approach we form two matrices that is BIBC matrix, which converts your branch bus injection to branch current and, then there is BCBV matrix which converts your branch current to bus voltages.

Once you get this two matrices, we multiply them get DLF matrix which is called as distribution load flow matrix and, from that matrix we try to follow the steps, which we which I have described in the last class. In last class basically we have seen how to form this BIBC and BCBV matrices.

(Refer Slide Time: 01:20)



So, in particularly we have seen; first say you are having the system which is having N buses and, you are having say in between there are l m n buses. And, it is want to write your BIBC matrix, which is basically on row side we have seen that we have to write the bus numbers, and here vertically you have to write branch numbers.

So, branch numbers here will be starting from 1 to 2, 2 to 3 like this say m nth branch and, then there will be last N minus 1 to Nth branch. And this will give us our BIBC matrix and, here the bus numbers are stating from 2 3 any bus m n and then say last bus your N bus.

So, in this case you have seen when actually taking first line it will terminated as a your first column of this BIBC matrix. So, whenever this line is coming we need to put 1 here. So, 1 2 we will get in the row of 1 2 we will get 1, everywhere actually we will get 0. And then we have see that we have to go on adding, you have to go on adding branch one by one.

Like we have seen if you are adding branch number 2 3. So, you have you can say that this column corresponding to bus 2 copied as it is and paste that another location that is 3. So, I am just copying this column as it is and I am testing at the location of 3. And then the entry corresponding to line 2 3, we have to make it 1 so, this particular entry we have to again make 1. So, this is actually 2 step algorithm first we have to code depending upon on which buses this line is connected we have to copy that particular bus and paste at particular location and, then you have to in second step you have to modify of entry corresponding to that particular line.

So, in generalized way if you are finding your m and nth column of this one, means if you are considering m nth line any m nth line, then it starting node is n and ending node is n. So, copy the column corresponding to m node and paste it at n location. Suppose this is your column corresponding to m node copied as it is and paste at nth location so, it will be same. So, I am just copying and pasting and then in second step you have to make m nth or your lined corresponding element of this matrix you have to make it 1. And you have to keep on repeating this process till all the lines are considered. So, if we considered all the lines then automatically your BIBC matrix it will get formed.

Similarly we have seen how to form BCBV matrix also, in case of BCBV matrix on row side you have to take branches. So, here the branches are 1 to 2, 2 to 3, 3 to 4 like this any branch m nth and then N minus 1 to Nth is the branch. And then on column side you have to take buses so, buses are 2 3 any m bus, any n bus and then up to N and then this will form your BC BCBV matrix I had written V so, it will be BCBV matrix branch current to bus voltages.

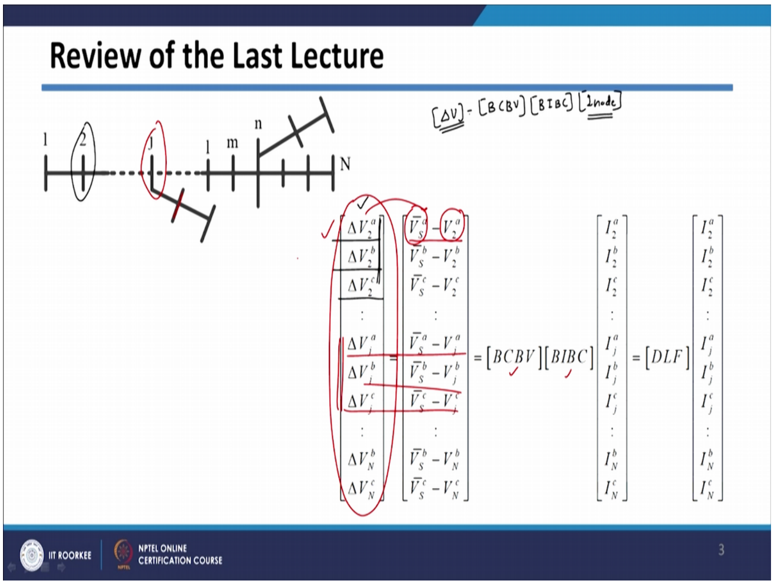
So, in this case first when you are starting when you are adding first line, you are actually add the entry corresponding to 1 2, you have to add the impedance of it and, then other rows other columns will be 0, when I am considering say line 2 3. Then what you have to do is you have to copy row corresponding to and paste at third row. So, copy this row corresponding to and copy a third location so, it will be Z 1 2 0 0 0 0 and then entry corresponding in the column of 2 3 and 3 corresponding to 3 you have to make it equal to impedance of that line.

So, this particular entry we need to make it, Z 2 3 like this if you want to add any branch which is a m nth branch, then you have to copy mth row because you are so, this is mth row and this is nth row you copy mth row of this matrix which is already formed. So, this get this mth row copy it and paste it at nth location as it is so, this is n you first step and your next step is entry corresponding to your m nth branch in this particular row, you have to replace it by impedance of Z m n.

So, this particular entry will be Z m n and you have to go on repeating this process, till all the branches are considered. So, in this case we will get all the rows of this matrix filled with entries corresponding to particular impedances. So, once you form this BIBC and BCBV matrices, we know that we calculate your DLF matrix, which is basically your BCBV matrix multiplied by your BIBC matrix. And as I told you as long as your of configuration of your distribution system is not changing, your DLF matrix will remain constant. So, you do not have to change this DLF matrix at every iteration.

Then we also seen how to form the BIBC and BCBV matrices for three phase system, the process is similar. However, in case of unbalance system where we reconsidering full three phase load flow analysis in that case so, all the three phase voltages will come into picture.

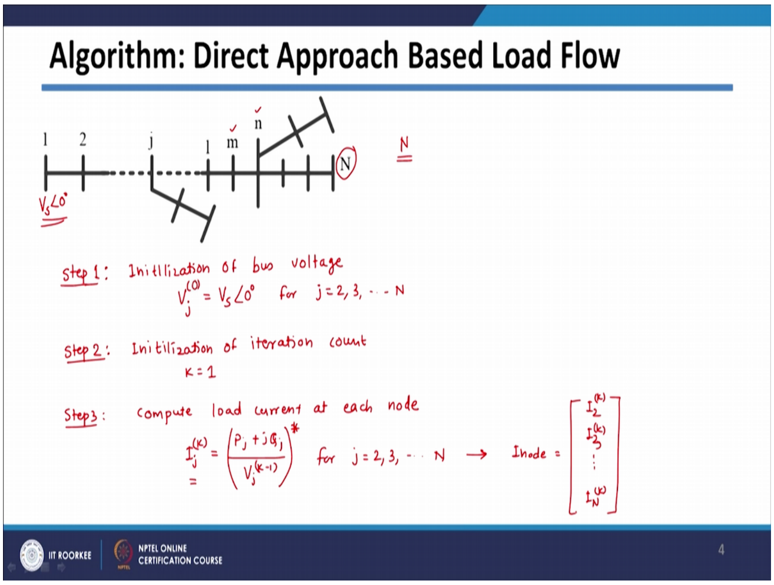
(Refer Slide Time: 08:25)



So, here your delta V which is basically equal to your BCBV matrix multiplied by BIBC matrix multiplied by your nodal currents. So, here you will get all three phase nodal currents and from that it will get actually voltage difference of all the nodes all the for the all the three phases. So, specifically your matrix of delta V will be something like this that is delta V 2 for a phase, delta V 2 for b phase, delta V 2 for c phase, particularly there are 3 voltages corresponding to this node number 3. And so, we know the delta V 2 is nothing, but your voltage difference till bus V 2 with respect to your starting bus.

Similarly, for any jth bus which is having all the three phases, you can get voltage difference of three phases and this BCBV matrices and b BIBC matrices will depend on how many nodes are there, or the how many three phase nodes are there. And your process of forming BCBV and BIBC matrix will remain same, we have already discussed in the last class. And once you solve it you will get actually delta V for each of the node.

(Refer Slide Time: 09:55)

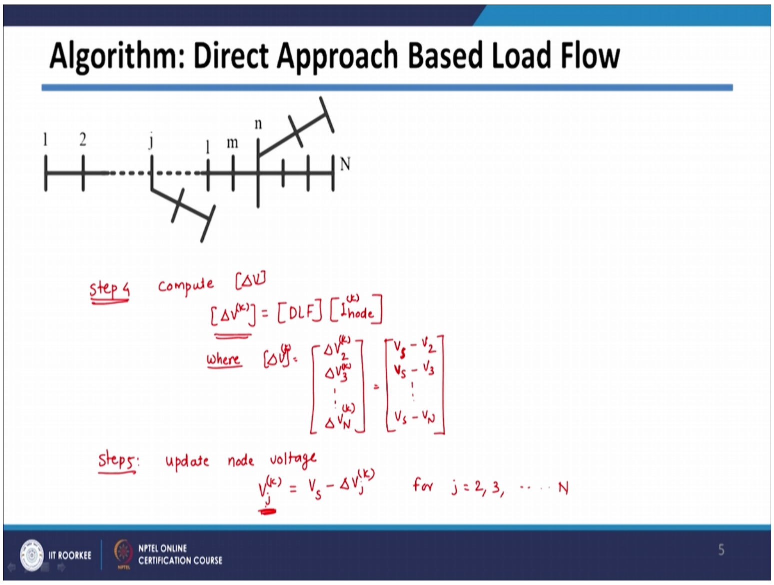


So, now may discuss the step by step algorithm of this BIBC based or BCBV based direct approach. So, in this case let us see the steps by steps. So, I am considering this system which is having say N number of buses in the system. And this is any m n nth bus I am considering, there are branches into the system. So, step 1 of this even if you are considering balanced or unbalanced system, step 1 will be initialization of bus voltages. So, it will be initialization of bus voltages so, all the bus voltages will be initialized to source node basically where the voltages known.

So, basically I know this voltages here which is V S angle 0 degree and all the buses will be initialized to that. So, all the buses say any jth bus will be initialized to V S angle 0 degree and, it is for all buses, where j is ranging from 2 3 up to N. So, I will initialize all bus voltages to V S angle 0 degree. And then step 2 will be initialization of iteration count, because you have to do this load flow iteratively. So, we need to initialize the iteration count to K is equal to 1. Step 3 will be calculation of nodal load current compute load current at each node and, we know that this load current will be calculated using this equation.

So, at any kth iteration your load current will depend on your load at that particular bus jth bus divided by it is voltage. But that voltage is available is only for earlier iteration that is why K minus 1 we need to put here, and then you have to takes complex conjugate for it. And it will be for all j-s ranging from 2 3 up to N. So, here I will get all the load currents at all the buses. So, from this I can form I node matrix here itself so, your I node matrix will be consisting of your entries, which are I 2 which I have calculated a kth iteration I 2, I 3 at kth iteration like this I will get in at kth iteration. So, this is nothing, but your I node matrix.

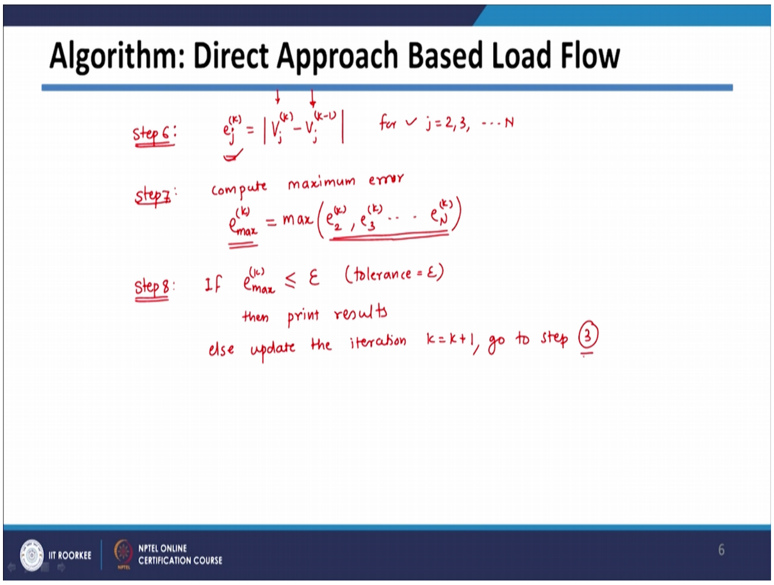
(Refer Slide Time: 13:03)



So, in step 4 we need to calculate delta V and, we have seen that your delta V matrix at any kth iteration will be equal to your DLF matrix, which is not going to change throughout the iterations it will remain constant multiplied by your I node current, which we have calculated at kth iteration, which will give me voltage difference at this one. And we know that voltage difference so, this matrix where your delta V is nothing, but delta V 2 delta V 3 up to delta V N and, which you have got at kth iteration. So, can just write that at kth iteration and we know that this is nothing, but your V 1 minus V 2, or it is V S minus V 2 and, then V S minus 3 and up to V S minus V N. And these are actually phasor quantities

So, once I get delta V in step 5, we get update the voltages so, update node voltages so, we know that your V j at kth iteration will be equal to V S minus delta V j which you got at kth iteration. So, this is basically will give you node voltages of all the nodes so, it is this step should be ranging for j is equal to 2 3 up to N. So, this how will get node voltages of all the nodes of kth iteration. .

(Refer Slide Time: 15:21)



Then 6th step the step number 6 we need to calculate error in the voltages so, error of jth node and kth iteration will be equal to your magnitude of kth iteration minus k minus 1th iteration. So, just we have subtracting voltages, which you have got in these iteration in earlier step and, the voltages which you have got in earlier iteration, which is k minus 1, we are subtracting them and we are seeing how much error we are getting into this one.

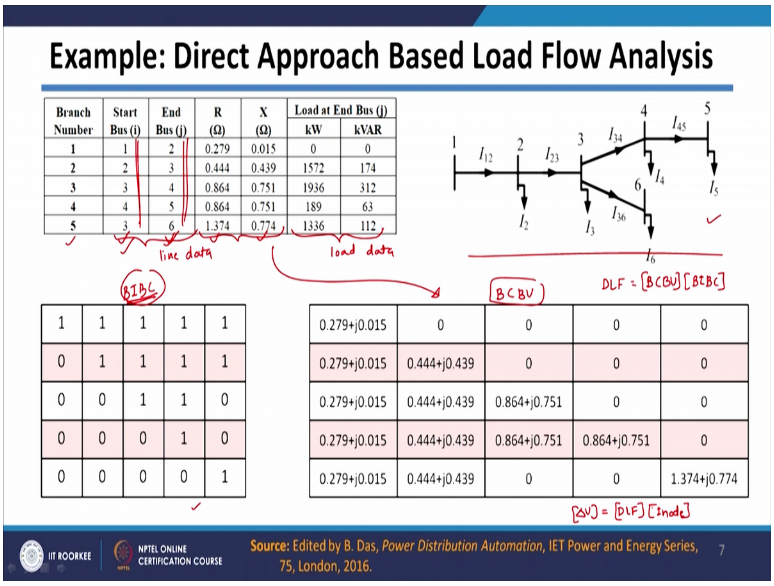
And then 7 is compute maximum error. So, maximum error at any kth iteration will be equal to max of so, here we are getting all the of errors of all the buses. So, did they also range from j is equal to 2 3 up to N, basically you are getting all e 2, e 3, e 4 and from that you have get maximum value.

So, e 2 at kth iteration, e 3 at kth iteration up to e N at kth iteration so, this will basically give this step it earlier giving you errors in all the buses, and we are calculating maximum error at particular bus. Once you get this maximum error compare this maximum error with respect to your tolerance value. So, your in this case you need to take, if your e max at kth iteration if it is less than your tolerance value. So, here the tolerance is epsilon.

Then if this is getting satisfied your converge you have converges load flow and, in that case you print the results. Otherwise else update the iteration count so, I will make if iteration count k is equal to k plus 1.

And go to step 3, because you already not converged here and then using the updated voltages, go to step 3 again calculate your currents of next iteration, calculate the voltage difference, or delta V for that iteration update the voltages, again calculate error maximum error and compare with your tolerance value. And if again results are not satisfied, then again go for step 3 and keep in this loop till we get converge results.

(Refer Slide Time: 18:32)



So, here this is you algorithm for direct approach based load flow method. Now, let us take one example which illustrate this particular approach of finding load flow solution. So, here I am taking the similar example which you have consider in case of backward forward sweep algorithm.

So, here as I explain there, these are the branch numbers, these are actually starting and ending nodes of that particular branch. These are resistance and reactance of that branch and this gives you the load data so, this is your load data and this is your land data. And as I told you using line data only we form your BIBC and BCBV matrices, as I told you we need only this two columns to form your BIBC matrix. So, by knowing the connections of nodes, which are which is given into this two columns we can form this particular BIBC matrix.

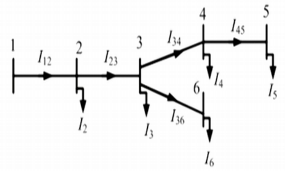
So, if you write BIBC matrix based on algorithm which is explained, in last two lectures we can get BIBC matrix for this particular system or for this particular diagram, which will be like this. Similarly we can form BCBV matrix and, for this forming BCBV matrix, we used this two columns as well as this impedance column here, because BCBV matrix consist of your impedances.

So, for this particular example your BCBV matrix will look something like this. And as I told you if you multiply your BC BV matrix multiplied by your BI BC matrix, which will give me DLF matrix. A from this DLF matrix we know that delta V voltage change, will be voltage difference with respect to source node, will be equal to DLF matrix multiplied by your I nodes.

(Refer Slide Time: 20:40)

Example: Direct Approach Based Load Flow Analysis

Branch Number	Start Bus (i)	End Bus (j)	R (Ω)	X (Ω)	Load at End Bus (j) kW	Load at End Bus (j) kVAR
1	1	2	0.279	0.015	0	0
2	2	3	0.444	0.439	1572	174
3	3	4	0.864	0.751	1936	312
4	4	5	0.864	0.751	189	63
5	3	6	1.374	0.774	1336	112



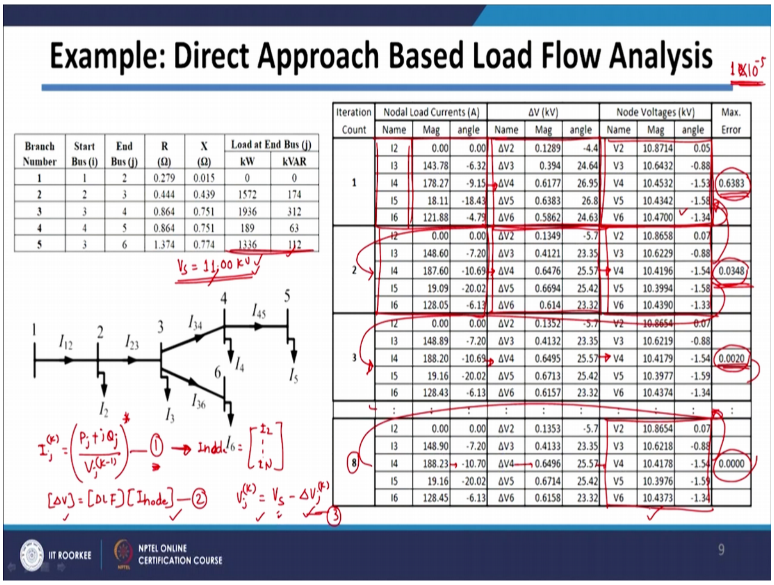
→[DLF] = [BCBV][BIBC]

0.279+j0.015	0.279+j0.015	0.279+j0.015	0.279+j0.015	0.279+j0.015
0.279+j0.015	0.723+j0.454	0.723+j0.454	0.723+j0.454	0.723+j0.454
0.279+j0.015	0.723+j0.454	1.587+j1.205	1.587+j1.205	0.723+j0.454
0.279+j0.015	0.723+j0.454	1.587+j1.205	2.451+j1.956	0.723+j0.454
0.279+j0.015	0.723+j0.454	0.723+j0.454	0.723+j0.454	2.097+j1.228

IIT ROORKEE | NPTEL ONLINE CERTIFICATION COURSE 8

So, let us see by what we get. So, here is your DLF matrix so, DLF matrix this is basically as I told you if you multiply this BCBV matrix which you got an last slide, multiplied by BIBC matrix you will get basically this DLF matrix.

(Refer Slide Time: 21:01)



And if you apply your load flow algorithm based on discussed, step by step process we get this results. Let us see what result we are getting it here, so initially in first step as I told you we assume load voltages. So, here your V S is actually 11 kV. So, in first step all the bus voltages will be considered 11 kV and, then we know that first equation which we used is actually calculating of node current. So, I j at any kth iteration, we have seen P j plus j Q j divided by V j at K minus 1th iteration and we have taken star of it. So, I am calling this is equation number 1.

Then we from this we have got I node. So, I node matrix we have seen it is matrix of I 2 up to I N and, after forming this I node matrix we have got delta V matrix, which is basically your DLF matrix multiplied by your I node which you have got it here. That, let us say this is equation number 2 and then we are getting V j at kth iteration which is nothing, but V S plus sorry minus your delta V j, we j getting at same iteration let us call it iteration number 3.

And this is all the algorithm is used here so, by using 11 kV we can using an equation number 1. Here we can get all the load current from the loads which are mentioned in this particular load data column, using these loads, using this voltage and using this equation number 1. We can get all the load currents from load currents get your delta V voltages, voltage difference with respect to source node. So, voltages voltage difference of all the nodes with respect to source node, will be calculated using equation number 2.

And if we calculate for this system in first iteration will get this delta V S and, then from equation number 3 by subtracting that delta V from your V S, V get node voltages so, here we got node voltages.

Now, calculate error because earlier iteration you are considering 11 kV as voltage of all the buses, and here we are getting some different voltages and get the error with respect to all the nodes. And then get maximum error and in this case we get maximum error is which is 0.6383, but our target of the error or tolerance which you have consider is 1 e exponential 5, or 1 into 10 raised to minus 5.

So, your error should be below this value so, here it is not satisfying so, we go to iteration number 2 again using these voltages which have got earlier iteration, we can calculate your nodal currents. From that again using equation number 2, we can calculate voltage this difference of the particular node with respect to source node, add that voltage difference, or subtract that voltage difference from your source node, to get the actual node voltages.

And then again with respect to earlier iteration voltages get the maximum error. So, here also the maximum error it is not getting satisfied and you have to go for iteration number 3 and, your process will keep on repeating get again nodal current delta V and new voltages calculate your error.

And if it is not satisfying again go to the next step and, if you do this for around 7 to 8 steps, you will come across your satisfied results means, if I subtract the voltages which you have got at 7th iteration from this 8th iteration I am getting error which is below 1 exponential minus 5. So, here I can say I have converged, this load flow solution and these are my final results of load flow. So, here we are getting this results of final voltages of all the buses which are converged 1.

So, this is how your direct approach based algorithm work. So, in today's lecture, we have started with writing the algorithm to get BIBC and BCBV matrix in generalize way, earlier we have seen particular case, but generalize way how can write if there is m nth branch.

If you are adding how your matrix will get changed and, in that case we can actually build your mat lab code or C plus code, or C code based on the algorithm, which I explained to from BIBC and BCBV matrices and, you can actually form BIBC BCBV matrices for particular distribution system. And once you get these BIBC and BCBV matrix. You get DLF matrix and as I told you DLF matrix will not change throughout the iteration and, then you can go for your direct approach based algorithm.

And then I explained you step by step process, basically there are 8 steps to get your load flow iteration to run. And then we have considered one example, in which I have illustrated, how this direct approach based algorithm work. So, for that particular case we have form BIBC matrix, BCBV matrix and, then DLF matrix and then we have got the result for various iteration till we converge.

Till now we have seen algorithm which are used only for radial system, whether it is for unbalanced, or balanced both the cases we have seen. So, your backward forward safe algorithm, as well as this direct approach based algorithm seen only for radial system. However, to increase the reliabilities sometimes distribution system is operated, in weekly meshed mode. So, in next class will see how we can saw load flow, if there are weekly meshed distribution systems.

Thank you.