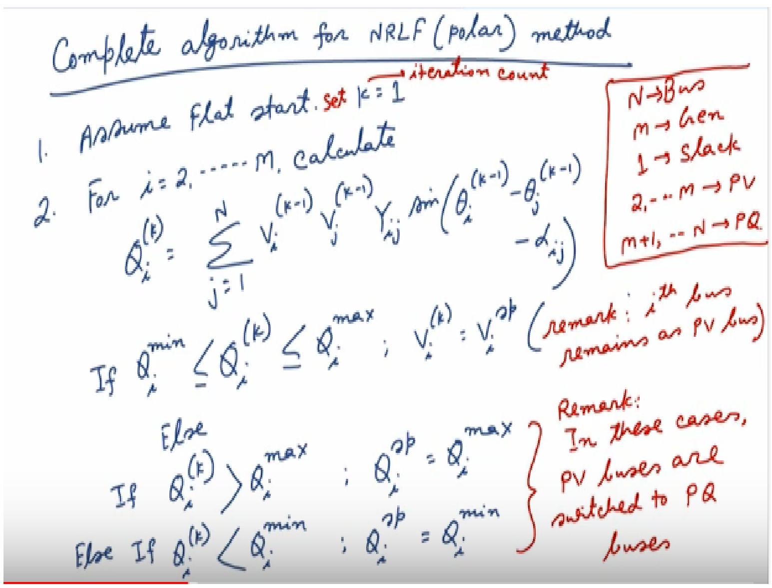
Computer Aided Power System Analysis
Prof. Biswarup Das
Department of Electrical Engineering
Indian Institute of Technology-Roorkee

Lecture - 15
NRLF (Polar) Algorithm and Example

Welcome to this module of this course computer aided power system analysis. After the last lecture we have derived the expressions of the elements of the Jacobian matrix. Earlier to that we have also looked at that what is meant by the correction vector and also what is meant by this mismatch vector. So now today, we would be looking at the complete algorithm of the NRLF polar method as well as one small example for this.

(Refer Slide Time: 00:59)



So complete algorithm. So in step 1 assume flat start. And we have already discussed what is meant by flat start when we also have generators in our system. Please do remember that we are considering a general system where you have got N- Bus M – Generators. Bus 1 is slack. Bus 2 to M PV M + 1 to N PQ. So this is our convention. So this for this where N and M can be of any value. So for this we are going to rather discuss this complete algorithm.

So the first one is the assumed flat start. Then what we do is then for i = 2 to M calculate Q i and let us say also initialize k = 1, k is equal to k is iteration count and we set k = 1 k is iteration count. So then we calculate Q i at kth iteration. So the first iteration as J = 1 to N V i (k – 1) V j (k – 1) Y ij sin (theta i (k – 1) – theta j (k – 1) – alpha ij). So what we are doing? For bus 2 to M we are calculating that what is the reactive power generated by this generators.

So what is 2 to M? 2 to M is in our parlance that we are essentially calculating that what are the reactive power generated by each of this generator at this present iteration. And when we are calculating the reactive power generator at kth iteration, in this case k = 1 when we start this kth iteration. At this point you have only got have the knowledge of the all bus voltage and angle corresponding to the last iteration. That is basically corresponding to the (k – 1)th iteration.

So then here in this expression in the last rather in this expression given at the right hand side we would be always using the values corresponding to (k – 1)th iteration. Then step 3, if Q ik is less than equal to Q i max and greater than equal to Q i min then we say that V i (k) = V i specified. Note the remark. It is for our own understanding. Remark ith bus remains as PV bus. If else if Q i k > Q i max then what will happen?

We have already discussed when we were discussing the Gauss – Seidel load flow analysis, we have already discussed that if this calculated reactive power by the generators remain within the corresponding limits of ith generator so then in that case that ith generator remains as a PV bus. On the other hand if this calculated reactive power crosses any of the limits either maximum or minimum so then in that case that particular bus voltage magnitude at V ith bus will not be maintained at the specified value.

It will be changed and that particular bus would be now treated as a PQ bus. Now in that case what will happen? Now because this calculated value of Q ik for example here in this case if it is crossing Q i max so then therefore physically this generator will not be able to supply this much of reactive power. It would be able to supply only a value equal to Q i max. So then in this case this particular generator will act as a PQ bus.

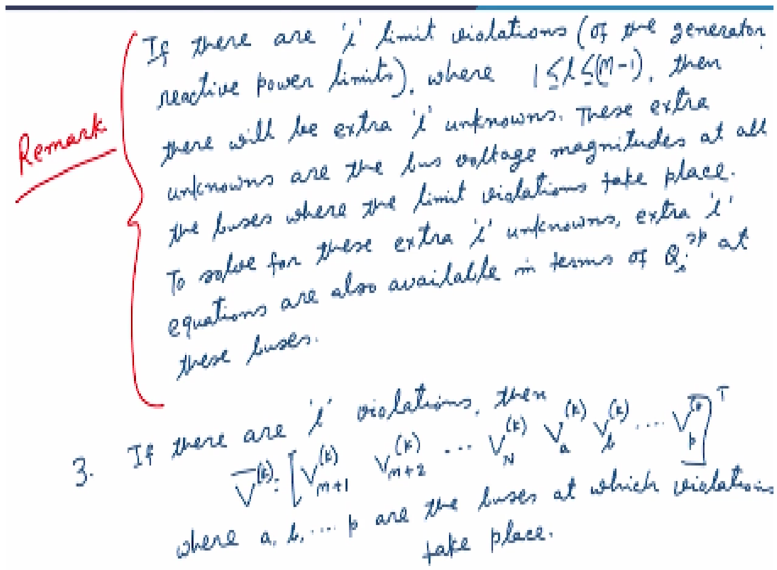
Where the specified value of Q which is being injected by this generator would be equal to Q i max. So then therefore if Q ik > Q i max Q i specified would be = Q i max and else if Q i (k) < Q i min then Q i specified would be = Q i min. Because in this case less than means that if this generator is asked to absorb more than this Q i min, so then obviously it would not be able to absorb more than this. So then it would be only able to absorb a value equal to Q i min.

So then in that case the specified value of the reactive power at the ith generator terminal would be equal to Q i min. We again take a remark for our own convenience. The remark is in these cases PV buses are switched to PQ buses. So again this is a remark. This is for our own understanding. Algorithm is only this. So then what would be the next step? Now here we need to understand one thing.

Suppose bus i has violated the limit. So then therefore what will happen? That means that particular bus i has now been changed to PQ bus. When we say that this bus has now been changed to PQ bus so then therefore its voltage magnitude will not be maintained at V i = V i specified. It would be something else which needs to be calculated. So then therefore for this particular bus when ith bus is being changed from PV to PQ one extra unknown is generated.

That is basically the extra unknown is corresponding to the its voltage magnitude. Now because there is one extra unknown is being generated so then therefore we also need one extra equation to solve for this extra unknown along with the other equations. Now from where we will get this extra equation? This extra equation we will get from this values of Q i specified is equal to either Q i max or Q i min, right?

(Refer Slide Time: 10:49)

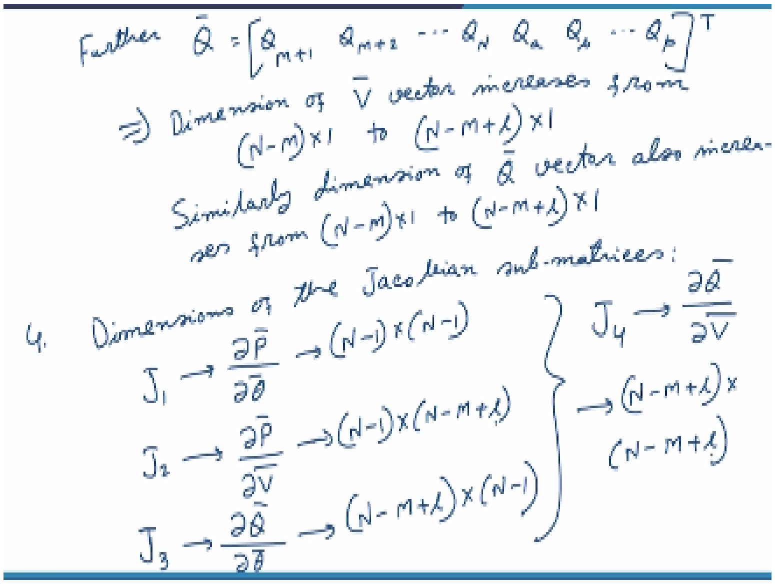


So then therefore we write that so it would be step 3. If there are l violations, l limit violations we write of the generator reactive power limits where l would be less than M and greater than equal to 1, less than equal to N – 1 because you see number of PV buses = M – 1. So then where l would be either greater than equal to 1 or less than equal to 1. Then there will be extra l unknowns. Now what would be these extra unknowns?

These extra unknowns are the bus voltage magnitudes we write in language to for better understanding. Bus voltage magnitudes at all the buses where the limit violations take place. So then therefore to solve for these extra l unknowns, extra l equations are also available in terms of Q i sp at these buses. In fact we should not actually write it as an algorithm step. We rather should write it as a remark. So we write it as a remark. It is also remark.

So now we come to step 3. In step 3 if there are l violations then what would be the V vector. V vector is already V M + 1, V M + 2 to V N and let us say this is V a, V b up to say V small p T where a, b, p are the buses at which violations take place.

(Refer Slide Time: 16:08)

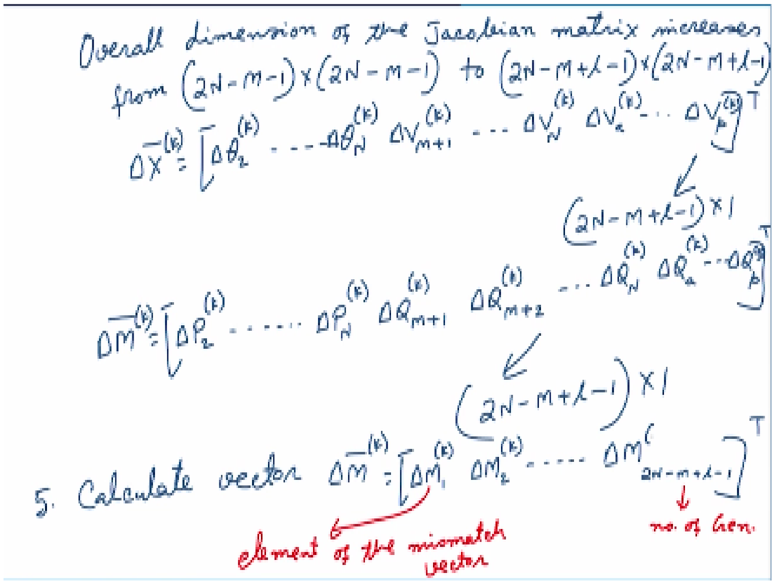


Further what is the vector Q? Q vector also changes. What is that Q vector? It becomes Q M + 1, Q M + 2 to Q N and then Q a, Q b, Q p. So then therefore dimension of V vector increases from N – M, earlier it was (N – M) * 1 to (N – M + l) * 1. Similarly, dimension of Q vector also increases from (N – M) * 1 to (N – M + l) * 1. Now what are the Jacobian elements.

So then what would be the Jacobian, so then dimensions of the Jacobian sub matrices J 1 is that is del P/del theta. It would be no change. It would be still (N – 1) * (N – 1) because the dimension of vector P or vector theta has not changed. What happens to J 2. That is del P/del V. What happens to this? It becomes (N – 1) * (N – M + l) because this vector V dimension has now changed from N –M to N – M + l. what happens to J 3? That is del Q/del theta.

Del Q has now increased. Now Q vector has now increased from or rather the dimension of Q vector has now increased for (N – M) to (N – M + l). So it becomes (N – M + l) * (N – 1) and what happens to J 4. That is del Q/del V. That becomes (N – M + l) * (N – M + l).

(Refer Slide Time: 20:13)



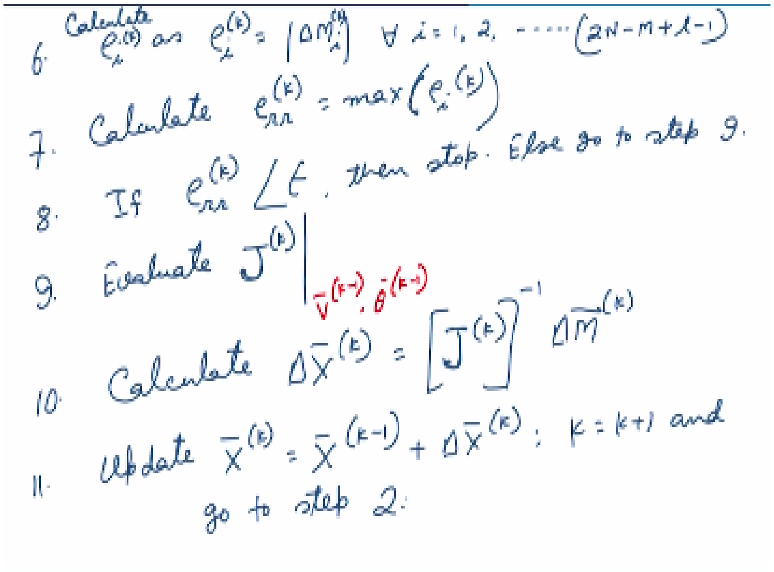
So then therefore overall Jacobian, so then overall dimension of the Jacobian matrix increases from (2 N – m – 1) * (2 N – m – 1) to (2N – m + l – 1) * (2N – m + l – 1). So the overall Jacobian matrix. So what happens to this delta X vector? Delta X. What is this delta X vector? Delta X vector is delta theta 2 to delta theta N and then it would be delta V M + 1 up to delta V N. Then delta V a up to delta V p. So this becomes this.

So its dimension also becomes (2 N – m + l - 1) * 1. What about delta M vector that is the mismatch vector? It becomes delta P 2, becomes delta sorry delta P N. Then delta Q M + 1, delta Q M + 2, delta Q N, delta Q a Q p transpose. This is also (2 N – m + l – 1) * 1, right. So then after that what we do? So once we do this, so once we do everything, everything is done. Now what we do is we calculate this vector delta M as defined above.

So we calculate all these elements delta M. And let us that this delta M vector is defined as let us say delta M 1, delta M 2 up to delta M 2 N – m + l – 1. Please note that this M and this M they are not the same. This M capital M denotes the mismatch vector or rather the basically the element of the mismatch vector and this m which is in the subscript they denote the number of generator. So we just write it.

So this is the number of generator in the system and this M all this big M’s these are the element of the mismatch vector.

(Refer Slide Time: 24:20)



So after that what we do? We calculate e i as e i = delta M i for all i = 1, 2 to (2N – m + 1 – 1). What we do is we simply take the absolute value of all of them. We just simply take the absolute value of this, absolute value of this, absolute value of this, just compute the absolute value of all the elements of the mismatch vector. We should write calculate. Then we calculate in fact everywhere it should be e i (k). It should be e i (k).

In fact everywhere it should be actually subscript k, k. everywhere there should be a subscript k because we are doing it at the kth iteration. We have missed it, k all are k. Here also it would be k. this is also k, k. These are all corresponding to kth iteration. These are the unknowns which we wish to solve for. These are the all unknowns. These are the knowns which we would be using. So these are all corresponding to k.

So then what we will do if then calculate error at k as max (e i (k). If error is less than some epsilon, stop. Else go to step 9. What we do in step 9? We simply evaluate the Jacobian matrix at kth iteration, evaluate J (k) using V vector (k – 1) and theta vector (k – 1). So evaluate J (k) by using this V N theta vector which is available in the last iteration and then we calculate delta X vector at kth iteration as inverse of so we take the inverse of this matrix.

Remember it is a matrix and it is a square matrix. So we take the inverse of this and we multiply this, we get this. And then we update X (k) = X (k – 1) + delta X (k). And we have already defined what is this matrix delta X (k) and X (k) would be theta 2 etc. k, update k = k + 1 and go to step, which step we will go back, step 2. So this is the complete algorithm. It is very simple algorithm, not really very much complicated.

Only thing is that at each and every iteration at the initial stage of this iteration we will first calculate that how many reactive power violations have taken place. So accordingly we will update our the unknown vector that is delta X. We will also update our mismatch vector that is delta M and we will also update our Jacobian matrix and we will also change their dimensions suitably.

And after that we will crosscheck that whether any of this mismatch has or rather whether all of this mismatches have gone down below certain threshold value or not. If they have gone down below certain threshold value so then we say that our algorithm has converged. Otherwise, we simply update rather after that we simply calculate the correction vector, update this voltage as well as angle and then go back and then reiterate.

So with this we will now look at a simple example. Simple example is that we again look at that same 5 bus system and in that 5 bus system if you do remember we had 3 generators and bus 1 generator is considered to be slack bus and bus 2 and 3 are considered to be the PV bus. So then therefore in this case N = 5 that is basically the number of bus is 5 and M that is the number of generator is 3. So then therefore J 1 is (N - 1) * (N – 1). So it is 4 * 4.

(Refer Slide Time: 31:09)

NRLF results in 5 bus system without any generator Q limit

$\mathbf{J}_1 \rightarrow (4 \times 4)$, $\mathbf{J}_2 \rightarrow (4 \times 2)$, $\mathbf{J}_3 \rightarrow (2 \times 4)$ and $\mathbf{J}_4 \rightarrow (2 \times 2)$.

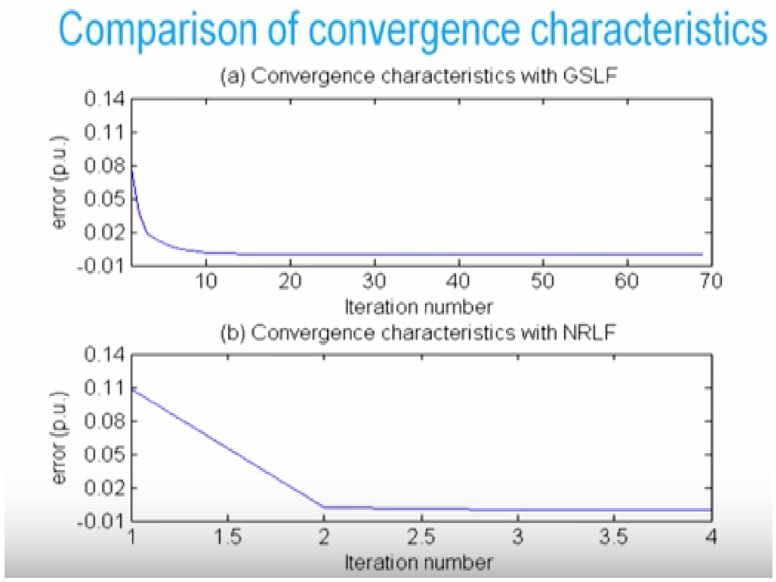
Bus no.	Without generator Q limit			
	$\lvert V \rvert$ (p.u)	θ (deg)	P_{inj} (p.u)	Q_{inj} (p.u)
1	1.0	0	0.56743	0.26505
2	1.0	1.65757	0.5	-0.18519
3	1.0	-0.91206	1.0	0.68875
4	0.90594	-8.35088	-1.15	-0.6
5	0.94397	-5.02735	-0.85	-0.4
	Total iteration = 4			

J 2 is (N – 1) * (N – M). So it is 4 * 2. Similarly, J 3 is 2 * 4 and J 4 is 2 * 2. And then we without generator Q limit that is we have initially considered that all this min and max values of the generator is pretty high. For example for all the generators this minimum value has been taken as -500 MVR. That is it can absorb up to 500 MVR and the maximum value has been taken as 500 MVR. That is it can generate up to 500 MVR.

So because we have taken a very large value of this limit so then it is virtually there is no generator limit. So then without any generator limit we iterate this and we find out this result. This result is identically the same as we have observed in the case of Gauss – Seidel load flow if you just compare it you will find it. Here again this bus voltage magnitude at bus 3, all this bus voltages are maintained at 1 and iteration is only 4.

On the other hand in the case of Gauss – Seidel load flow the number of iteration was much more. It was I think, it was something like 69. So let us look at the comparison of convergence characteristics.

(Refer Slide Time: 32:30)



So if you look at the convergence characteristic of GSLF, GSLF converging characteristic is very slow. It takes roughly about 69 iterations. On the other hand in the case of NRLF this convergence characteristic is very fast. It hardly takes 4 iterations. Please remember in this case we have taken this threshold value as 10 to the power -12. So here also we have taken 10 to the power -12 and for NRLF also we have taken 10 to the power of -12.

So with this GSLF takes 69 iterations and with this NRLF takes 4 iteration. So NRLF takes much less number of iterations as compared to GSLF. When I do the same exercise by applying this generator Q limit this generator Q limit is the same as we have considered for this GSLF. Then we find out this result. Again the result is also the same as we have obtained for the GSLF. And here also this particular bus voltage magnitude is not now remaining at, is not now remaining at 1.0 per unit. Because now here what happened?

(Refer Slide Time: 33:53)

Final Results of the 5 bus system with NRLF (polar) with generator Q limit

Bus no.	Without generator Q limit				With generator Q limit			
	$\|V\|$ (p.u)	θ (deg)	P_{inj} (p.u)	Q_{inj} (p.u)	$\|V\|$ (p.u)	θ (deg)	P_{inj} (p.u)	Q_{inj} (p.u)
1	1.0	0	0.56743	0.26505	1.0	0	0.56979	0.33935
2	1.0	1.65757	0.5	-0.18519	1.0	1.69679	0.5	-0.04769
3	1.0	-0.91206	1.0	0.68875	0.9825	-0.63991	1.0	0.5
4	0.90594	-8.35088	-1.15	-0.6	0.88918	-8.35906	-1.15	-0.6
5	0.94397	-5.02735	-0.85	-0.4	0.93445	-4.98675	-0.85	-0.4
	Total iteration = 4				Total iteration = 5			

For bus 3 we have now reduced this maximum reactive power generation limit as 50 MVR. But as we have seen in the case of GSLF this particular generator needs to supply more than 65 MVR to maintain its voltage at 1.0 per unit but because its generator production limit has been clipped at 50 MVR so it can only supply up to 50 MVR. So as a result its voltage has gone down. So earlier without any generator Q limit it was maintained at 1.0 per unit.

But now it is maintained at 0.9825 and rest of the other things have been and the other results are identically the same. Again, this total iteration has just increased by 1 5. So 4 and 5 there is not much of a difference. So even with generator Q limit the number of iterations taken by NRLF is much less than what is taken by GSLF. So in this lecture we have looked into the complete algorithm of the NRLF polar considering all this practical aspects that is the number of generators as well as their generator Q limits.

And we have also looked into and one small example that is the same 5 bus system example and we have seen that this results obtained by GSLF and NRLF are identically the same although the number of iterations taken by this NRLF are much less than that taken by GSLF load flow algorithm. So from the next lecture onwards we will be considering some other algorithms. Thank you.