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

Lecture - 11
Basics of Newton Raphson Numerical Method

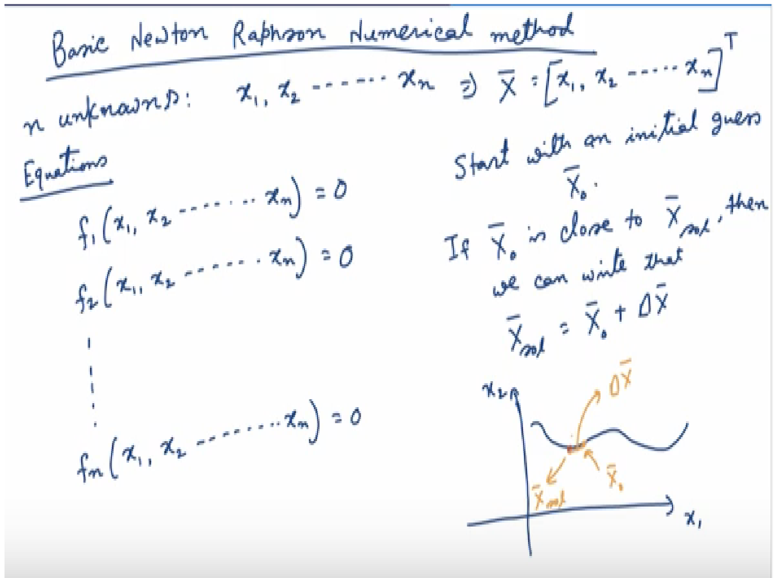
Welcome to this lecture of this course on computer aided power system analysis. In the last lecture we have looked into the example of the Gauss – Seidel load flow analysis for a small 5 bus system in which we have seen that Gauss – Seidel load flow technique takes a reasonably large number of iteration. For example without any generator reactive power limit this system was taking 69 iterations and with generator Q limit this system was taking 66 iterations.

So then we can imaging that if the number of iteration is I mean if the number of iteration is this much for a small 5 bus network so then obviously for any large power system for example say 30 bus, 100 bus or let us say 1000 bus systems the number of iterations taken by a Gauss – Seidel load flow analysis would be quite substantial. So as a result usually in the industry Gauss – Seidel load flow method is not very much preferred.

Instead of that in the industry usually or rather the so called de facto method of analysis of any large power grid is essentially based on the Newton Raphson numerical method. So from this lecture onwards we would be now discussing this Newton Raphson numerical method to all its detail. But then today in this lecture we will first look into the basics of the Newton Raphson numerical technique.

So that after we do understand the basic steps or rather the basic philosophy of this Newton Raphson numerical method we should be able to apply this basic steps or basic philosophy for the solution of the power flow equations. So today we start with this basics of Newton Raphson mathematical technique or rather Newton Raphson mathematical sorry Newton Raphson numerical method.

(Refer Slide Time: 02:56)



So today we look at the basic Newton Raphson numerical method. We must I mean we must mention here that this particular Newton Raphson numerical method is very powerful method and this method has got a very large applications across all branches of science and engineering. So then therefore we need to give a little more attention to the understanding of this very basic Newton Raphson numerical technique.

Because if you do understand this very basic technique so then we would be able to apply it without any problem not only for the solution of the powerful equations but for the solutions of any set of nonlinear algebraic simultaneous equations involving any number of unknowns. So then let us assume here that we have got n unknowns and these unknowns are x 1, x 2 up to x n. Now because there are n unknowns so we need to have n equations.

So equations are some equation f 1 (x 1, x 2… x n) = 0. Please note that we are not assuming any particular form of this equation. This equation can be a I mean this equation can have any form, right? This equation can have any number of terms of nonlinearity. It simply does not matter. So we are not assuming any particular form of this equation. This equations can be of any form. Only restriction is that these are nothing but the algebraic equations.

That is all, nothing else. So f n (x 1, x 2 to x n) = 0. Now in the Newton Raphson technique what we do is that we again because this is basically any kind of numerical method. So then what we do? That we first assume some initial point. I mean we have to solve for x 1, x 2, x n. So we call that this is a vector x bar. So this is an unknown vector x bar. Because this is a vector I should say capital X. This is a vector, so this bar is not needed.

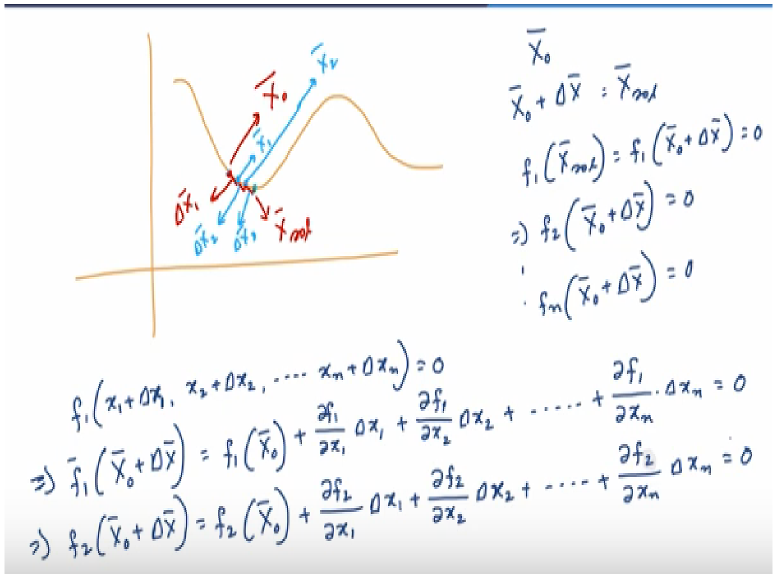
So this is my vector, unknown vector, x 1, x 2, x n. So this is a column vector. Now what we start, what we start is we start with an initial guess x nought. So this is a vector so let us put a bar. So we start with x nought. Now if x nought is sufficiently close to this actual value of x so then therefore you can write, if x nought is close to x 1 x. If x nought is close to x true or rather x solution; x solution denotes this actual solution.

Then we can write that x solution = x nought plus some delta x vector. So that means now what exactly we are doing? I mean what exactly we are doing. I mean because this is a let us say I mean there are n unknowns but to understand the basic philosophy let us take a very simple case of 2 unknowns; so x 1 and x 2. So let us say we are taking 2 unknowns. So x 1, x 2 and there is a curve, some curve.

And here we are interested to find out, let us say I am interested to find out this solution. So I am interested to find out let us say this is my solution. I am interested to find out this solution. Now I do not know what is the solution that is what I am trying to find out. But then I do have some idea about this particular solution, some idea that is in which zone it will reside and etc. I have got some idea. So then therefore I take an initial guess which is close to that.

So for example, so I can take an initial guess say somewhere here. This is my true solution. This is my x sol and this is my x nought, right? This is my x nought. It is actually, so now if I have to find out x solution from x nought, so then therefore if I can find out this difference, this difference delta x. So then therefore if I can somehow find out this difference delta x somehow so then I can simply add this delta x to this x nought and then simply reach at this value of x solution. So that is the basic philosophy. Now usually what happens? Usually what happens?

(Refer Slide Time: 09:45)



That say I am just taking a curve, something like this and I am interested say this is the solution. Now it is not always possible for us to find out a or rather to have an initial guess which is very close to this. So it may happen that, it may happen that my initial guess is somewhere distant as compared to this solution. So this is the x solution and this is x nought. So then what we do is, we can, so then what we do is that we simply try to reach that point.

So then therefore, so now our objective is to find out this gap. That is basically this gap, essentially this particular distance from x nought to x solution. But because they are now quite apart, so then it is not possible for me to find out this distance in one go. So then what we do is that we try to progress towards the solution in small steps. So that means I go into a very small step, I take a slope here. Take a small step and then by the slope I then reach here.

Then again I take a slope and again I reach here. Then I take again a slope and then again I reach here. So then therefore what we do is that we take some, we simply calculate let us say delta x 1. Then I so this is x nought. So then this point, so this point is let us say x 1. After this I calculate here, again I do this some calculation whatever is that calculation that we will actually we will now just now discuss. After that we calculate delta x 2, some distance. Then we proceed.

Then we reach this point. This point is x 2. And then we keep on doing this. Say this is after that I again do this delta x 3 and then we reach the solution. So what we have done here, essentially we have started with an initial guess and then we have calculated small distances and tried to approach my solution. Now how have we done this some small distance. Basically what we have done is, here we have simply calculated the slope at this point and then we have moved along the slope by a small distance and then updated my solution.

Then again when I reach at this point x 1 again I have calculated the slope at this point. And then again I have moved towards the solution and then again updated my solution vector and then again I have moved along with the direction of this particular slope at this point and then again updated the solution. So if I do this process, hopefully and if I can ensure that I will be reaching at this I mean if I can ensure that if that if we are moving into the correct direction, ultimately we would be reaching at the solution.

So then what we are doing? We are essentially calculating the slope at this point and then we are updating the our solution vector. Now, initially when we start, so then what we are saying that we are saying that let my solution is, let my initial guess is x nought and I assume that my solution is actual solution is x nought plus delta x. So my task is to find out this value of delta x.

So then therefore, if I assume that this is my solution, x solution so then therefore f 1 (x solution) that is f 1 (x nought + delta x) = 0. Similarly, f 2 (x nought + delta x) would be 0 and … f nought (x nought + delta x) would be 0. Now my objective is to calculate this value of delta x because we have already assumed x nought and if I am able to find out delta x so then I will simply add this value delta x and then find out the true solution.

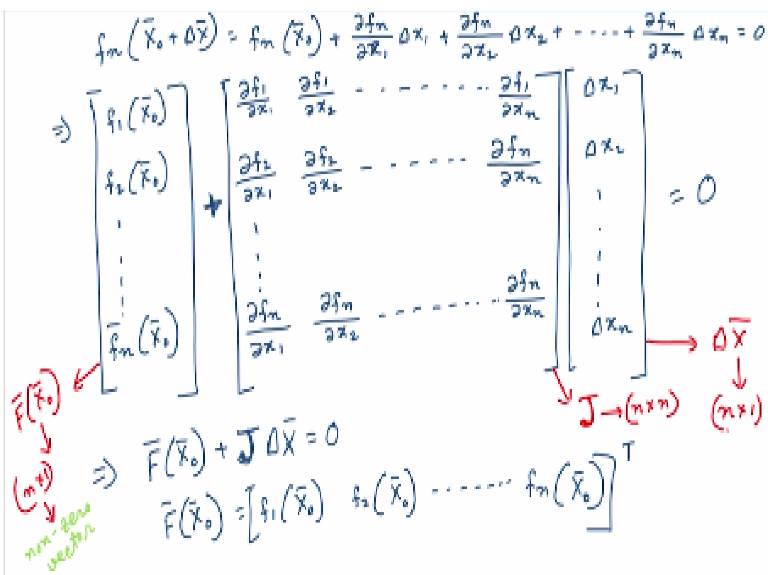
So now what I have is, so I have got f 1, so I am now expanding it. X 1 + delta x; x 1 sorry delta x 1 then x 2 + delta x 2…x n + delta x n) = 0. Now if these values are small, delta x 1, delta x 2, delta x n are small. That is if I have just travelled by a very small distance from let us say x nought with delta x. So then therefore for all practical purpose delta x 1, delta x 2, delta x n this particular equation can be linearized at this point. So then here what we do?

We simply apply the standard Taylor series expansion. So what is the Taylor series expansion? So we just apply Taylor series expansion. So in that Taylor series expansion what we have is f 1, so we would be now for the brevity x nought + delta x is given by this is the standard Taylor series expansion f nought (x nought) + del f 1 del x 1 * delta x 1 + del f 1 del x 2 * delta x 2 + … del f 1 * delta x n * delta x n = 0.

Here, please note that we have simply neglected all the higher order terms saying that, that these values of delta x 1, delta x 2 and delta x n are very small. So then therefore if you take this higher order terms so what I will have? We will have single delta x 1 square delta x 2 square delta x n square. Now because this value delta x 1, delta x 2 and delta x n are already small so if you take their square they would be even lower.

So then for all practical purposes they will be only neglected. So then this is the final expression. Similarly, for f 2 from this equation f 2 (x nought + delta x) = f 2. Again the standard Taylor series expansion delta f 2 delta x 1 * delta x 1 + delta f 2 delta x 2 + … del f 1 del x n delta x n = 0. Similarly, we do this operation for all this.

(Refer Slide Time: 18:06)

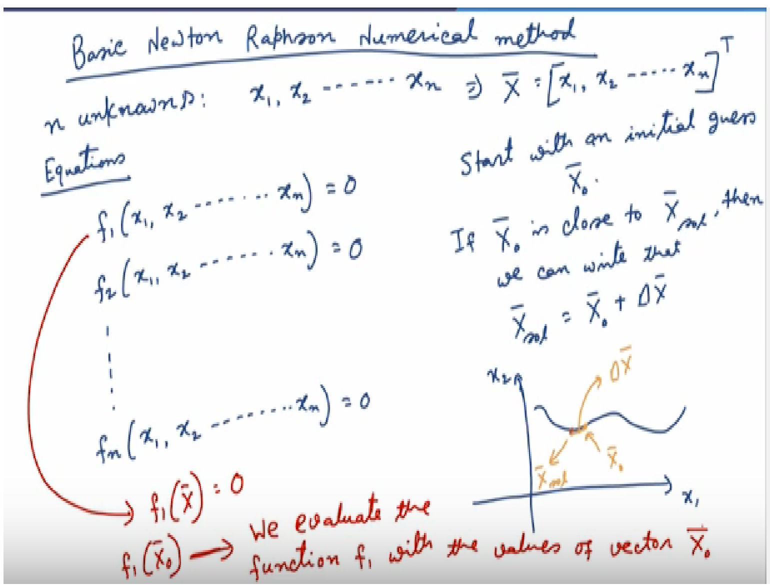


So f n (x nought + delta x) = f n (x nought + del f n x 1 * delta x 1. Del f n del x n delta x n = 0. So then therefore what we have? So now if we now collect all this equations together and write them in a matrix form so what I have? I have is f 1(x nought) f 2 (x nought)…f n(x nought) = sorry plus here it would be second row would be del f 2 del x 1, del f 2 del x 2, del f n del x n… then del f n del x 1, del f n del x 2, del f n del x n = 0.

So if we now write, so this is a matrix equation. So now if we do write down this matrix equation in a more (()) (20:42) form what I get is I write it as F, vector F (x nought) + this is called the Jacobian matrix, matrix J * vector delta x bar = 0. Where vector x x nought is = f 1(x nought bar) f 2(x nought bar)…f n(x nought bar) transpose. So we are actually denoting this vector as, so this vector as vector x which is a function of x nought.

This is delta x vector and this is the Jacobian matrix. Now what is the size? Size is n*1. This has also got size n*1. And this has got a size n*n. So this is a square matrix. Now in this expression what is meant by this f 1 which is a function of x nought. It means, basically it means that I have a function f 1, x 2, x n = 0.

(Refer Slide Time: 22:31)

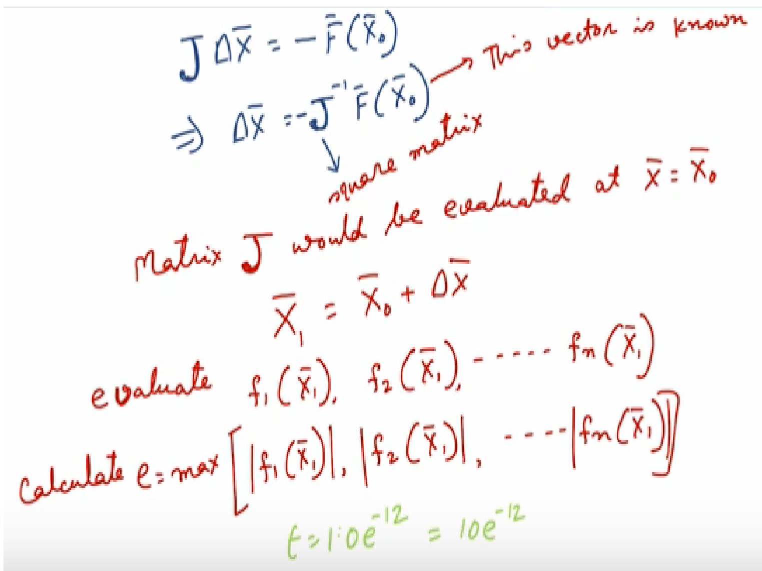


So when I say, so then here I can write down for example here so this equation for example can be written as f 1 is a vector of x = 0. Where this vector x is already defined. Now when we say that f 1 (x nought) it means that we evaluate the function f 1 with the values of vector x nought. So then therefore what we do? We simply take the initial guess x nought. So then therefore you know all the numerical values.

And we substitute this numerical values in this expression, whatever is that expression. Now because this initial guess is not the solution, so then therefore this expression when we do evaluate f 1 with this value of x nought this expression will not be equal to 0. Similarly, when I say that f 2 x nought that only means that we are evaluating the function f 2 with the values given by x nought and again as x nought is not the solution or rather is not the true solution so then therefore when we do evaluate f 2 with this value x nought this will also not be a 0 quantity.

Similarly, all the other functions when they are evaluated with the values given by x nought they would be nonzero. So then therefore in general, so then therefore this particular vector would be a nonzero vector. So it would be a nonzero vector. So it would be a nonzero vector. So f x nought would be a nonzero vector. Now what is about this Jacobian? Now we want to solve this delta x 1, delta x 2, delta x n.

(Refer Slide Time: 25:20)



Now from this equation what I can write is, from this equation we can write is that J * delta x bar = - F x nought bar. So then therefore from this it is very easy to find out J inverse into minus. Now because this is a square matrix and if this is an invertible matrix and or rather if this is an nonsingular matrix then it will be invertible. So this is a square matrix and if this is an nonsingular matrix it will be invertible.

Now the point is if we have to do this numerical analysis. Now remember this is known. This vector is known. This vector is known because we have evaluated them. So basically we have evaluated each and every function f 1, f 2, f n with this vector x nought and we have calculated their values. But how about this matrix J? From where we would get that numerical values of this matrix J?

So answer lies is that first we would calculate their analytical expression and then we will substitute the values of x 1 nought, x 2 nought, x 3 nought into each of these analytical expression and find out their numerical value. So then therefore what we write is that this matrix J would be evaluated at x = x nought. So then therefore what we will do? We will first find out the analytical expression of all this partial derivatives.

Because f 1 is a function of x 1 to x n. So then therefore this partial derivative or rather all this partial derivatives will also be a function of x 1 to x n. So then therefore we will be evaluating all this partial derivative functions with the values of x nought. So then therefore we would find out the numerical values of all this elements of this matrix J and then therefore we would be able to find out this numerical matrix J.

Once we find out this numerical matrix J we can simply perform this numerical calculation. So from here we would find out delta x. Once we find out this delta x then what we will do? We update our solution x 1 as x nought + delta x. So this is my new solution. So now we have to crosscheck whether this is my true solution or not. So then what we will do is we will evaluate all this functions, evaluate all functions with x 1, right?

Now if x 1 is really the true value, true solution that is if we have been able to find out this particular vector such that x 1 is really the true solution so then therefore f 1 (x 1), f 2 (x 1), f n (x 1) would be identically 0. But in practice because there are numerical values if x 1 is very close to the true solution then this value would be almost equal to 0. So then therefore the difference between 0 and this numerical value would be pretty small.

So then therefore to check whether this x 1 is really the true solution or not what we do is we simply calculate the magnitude of all this quantity, magnitude. We calculate magnitude. We calculate magnitude of all this quantity, right and we take their maximum value. Calculate error e

= maximum of this magnitudes. So this is also magnitude, this is also magnitude. So we calculate all their magnitudes.

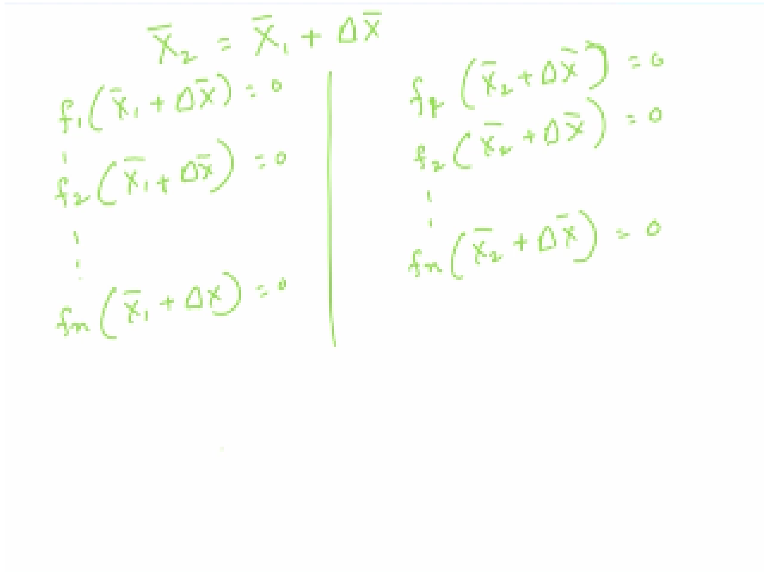
Now if x 1 is the really the true solution so then therefore their magnitudes would be almost equal to 0. All these values I mean all this functional values would be almost equal to 0. So then therefore if we take the maximum of them also, so then therefore this maximum quantity would be also almost equal to 0. So then therefore if that maximum quantity is less than some convergence value or rather less than some particular threshold value, that is very small value.

So then we can say that for all practical purpose these values are almost equal to 0. So then therefore if we take our say our epsilon to be as we have discussed that 1.0 * e to the power -12. That is equal to 10 to the power * e to the power -12. So then therefore if this value of e is less than this value, so then therefore this value of f 1 evaluated at x 1 is different from 0 only after 13th decimal point.

Similarly, the value of f 2 evaluated at x 1 is different from 0 is only after 12th or 13th decimal point. So then therefore because their value is different from 0 only after 12th or 13th decimal point so then therefore for all practical purpose as calculation we can take their values to be 0. That means we have found out the solution. If not if these values, if this maximum value is less than this, I mean if this maximum value of e is more than this.

So that means there is at least one value, there is at least one functional value which is not different from 0, right? Or rather which is basically different from 0 significantly. So then therefore we have to reiterate it. So then what we do? So if this value of e is still more than e or rather still more than epsilon so then what we say that there is at least one functional value which is still significantly more than this 0 value. So therefore we have to reiterate. So then what we do?

(Refer Slide Time: 33:16)



We again assume that let my true solution is let us say x 2 and I again do this delta x for the next iteration and then we again do the same thing with this value and keep on doing the same thing till and till this value of delta x becomes very small. So then what we do is after we get this, if this value of e is more than e so then we again evaluate x 2 or rather then again we do the same thing that is we again evaluate at x 1 + delta.

Let us say x 1 f 2 at again we evaluate it delta x again we evaluate f n by doing this Taylor series expansion etc. And then again we again recalculate this values of delta x again and then again we update x 2 as x 1 + delta x. Again check whether any of this functional value evaluated at x 2 is significantly more than 0 or not. If it is not, then we converge. If it is then we again do the same calculation at x 2 + delta x sorry this is f 1, f 2 at delta x … f n (x 2 at delta x).

And we keep on doing this iteration till we reach the convergence. So this is the basic Newton Raphson method and we will apply this basic Newton Raphson method for the solution of the power flow equations in the next lecture. Thank you.