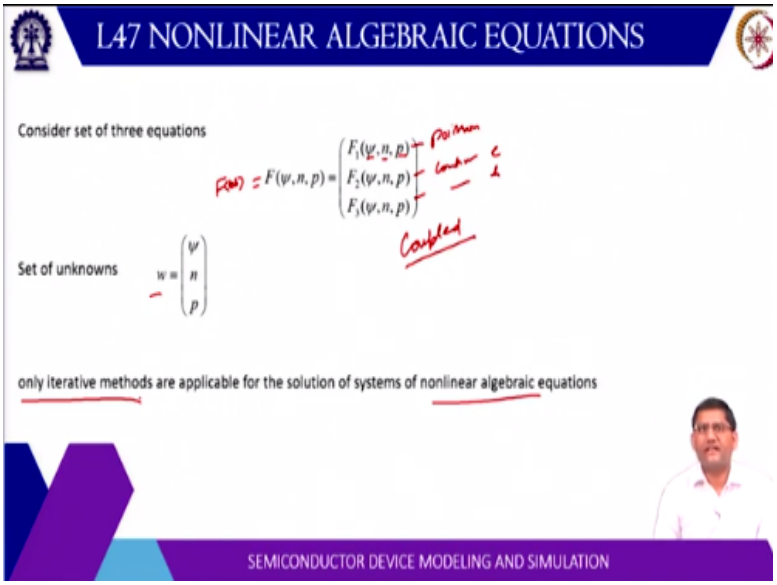
Semiconductor Device Modeling and Simulation
Prof. Vivek Dixit
Department of Electronics and Electrical Communication Engineering
Indian Institute of Technology-Kharagpur

Lecture - 47
Solving DD Equations (Contd.)

Hello, welcome to lecture number 47.

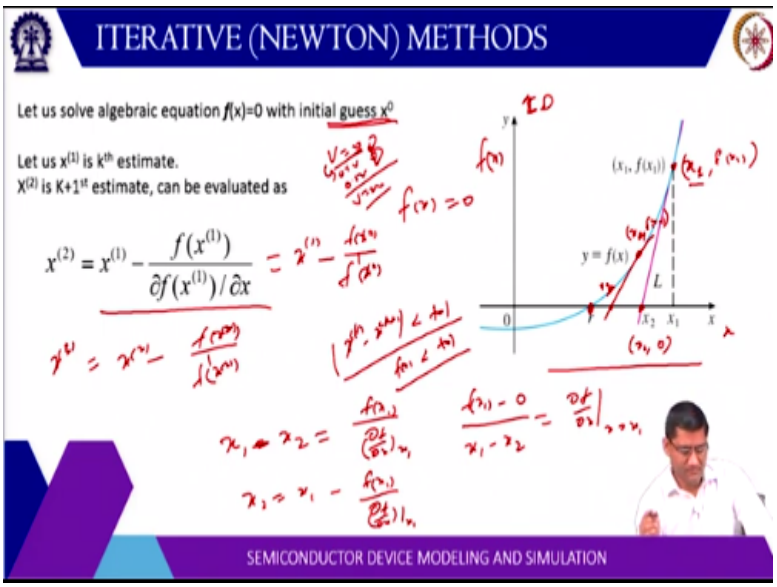
(Refer Slide Time: 00:27)



In this lecture, we will discuss the solution to nonlinear algebraic equations. Now let us consider a set of three equations. So as we have discussed, there is a Poisson equation, then there is a continuity equation for electron and continuity equation for holes. So these are the three equations. And if you notice, these three equations are not equation of one variable but they all involve the three variables.

So we can say these three equations are the coupled equations, three equations are the functions of three unknowns psi, n and p. So we can again represent the set of unknown as w, and this is a vector psi, n and p and it can also be written as F(w) to solve this set of coupled equations. So coupled nonlinear algebraic equations, only iterative methods can be used. So in iterative method, what we do?

(Refer Slide Time: 01:30)



They are some variation of the Newton method basically. Now Newton method what it does? We begin with some initial estimate. So let us say initial guess I am using this 1D for illustration purpose, okay? So let us say there is a function f x, which is a function of x, and we have to solve this equation f (x) = 0. So that means we are here to find the root of this function where it crosses at this x is x equal to zero.

So we start with some initial guess. Let us say this is your initial guess x 0 and or let us say this is x 1 initial guess. Now from x 1, the next guess should we nearer to this route basically. So what we do, we assume that this is a kind of straight line. So we take the tangent of this function at the point x 1. So if you write this equation of this line, so this point is x 1, f (x 1) and another point here will be some point x 2 which we have to find and with value y is zero.

So f x is 0 here. Now if you write the equation of this line, you can write f (x 1) minus 0 divided by x 1 minus x 2. This is basically the derivative del f by del x at x equal to x 1. So we simplify this one, what you get? x 1 – x 2 is equal to f (x 1) divided by del f by del x at x 1. So from here you can find x 2 is equal to x 1 minus f (x 1) divided by del f by del x at x 1.

So the next x estimate will be x 2. So x 2 will be x 1 minus f (x) by it is simply written as x first iteration minus f (x 1) divided by f prime x 1 to get the x 2. Now next estimate will be x 3, which will be x 2 minus f (x 2) divided by f prime x 2. So

derivative of f. So this is x 2. Now so next we will evaluate again x. So here we get x 2, f (x 2). And again we do the same thing.

So here we can draw. So this will be your x 3 and then again we do. So we will by this repeated process, we will come sufficiently close to this route r. And when we say that the error that x 1 minus let us say for k-th iteration x k - x (k+1) is smaller than certain tolerance, right? We say it has converged and value of fx is also smaller than some tolerance. Then we say that this has converged basically.

Now as you can easily identify that this actually gives a value which is close to you know the closest route because it is possible this function may have multiple routes. So if you are close to certain route it will converse to that route provided there is no inflection point. So there are lot of you know mathematics involved and theorems, so which we will not cover, but we will just understand what is this process basically.

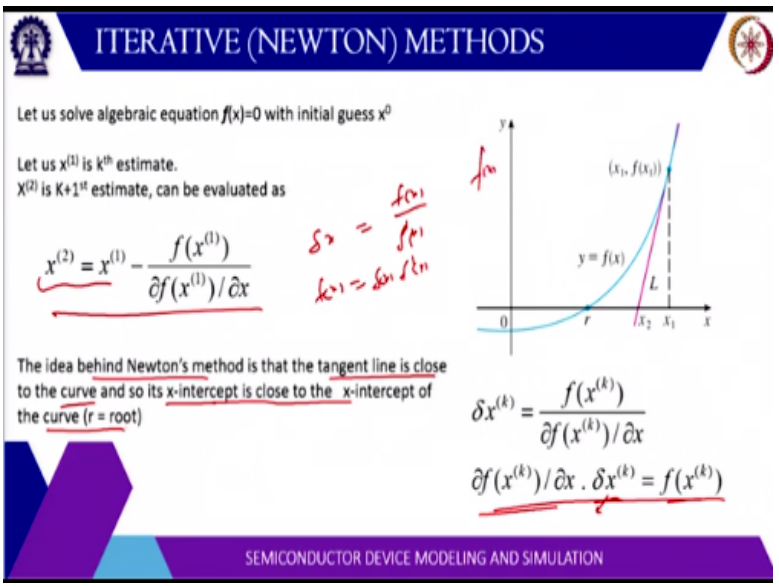
So we assume that your initial guess x naught is sufficiently close to the solution and when you go ahead with this method it actually converges. So if you take any numerical methods book, there will be some example related to this Newton method and in some cases it converge and the function is not appropriate it may not converge.

But as far as semiconductor device simulation is concerned, we frequently use it because we fairly know where to expect the solution of given set of equations. Then what we also do, we start with let us say some potential V applied equal to 0. So that is the equilibrium solution. So that becomes your guess.

Then you increment this del voltage by let us say 0.1 volt or 0.5 volt, and then gradually we increase and this first solution x as the initial guess so that it fits. So that way we can get the solution for even high voltages, because we are not abruptly changing the voltage. If you abruptly change the voltage, let us say you solve for V equal to 0 then V equal to 2 volt, it may not converge.

So we have to go step by step. So Newton method converges if the guess, initial guess is sufficiently close to the solution.

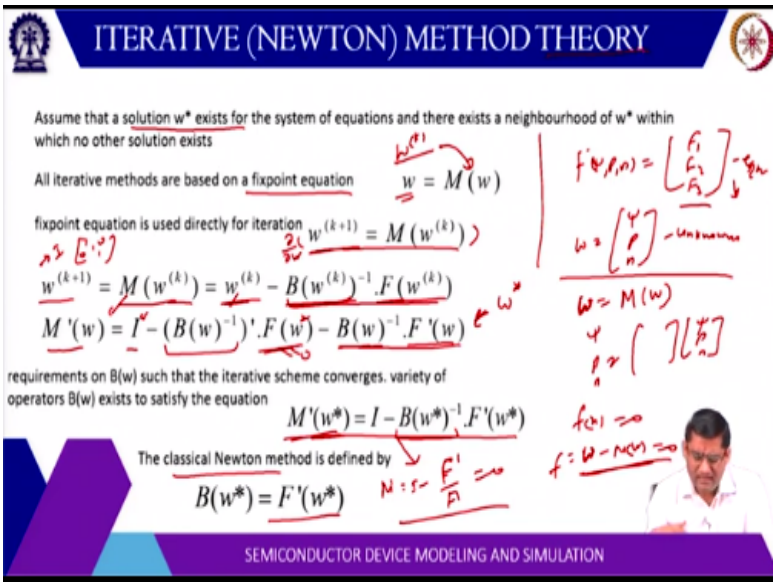
(Refer Slide Time: 07:00)



Now let us look this for 3D. So this is the same thing, I have already discussed. So del f by del x delta x k is x (k+1) - xk right. So this can also be written as, so x 2 minus x 1 is delta x is equal to fx by f prime x. So your fx is basically delta x times f prime x. So this one here f prime x times delta x is equal to fx. So this is how, this is basically calculated and this can be applied for higher dimension also okay.

So the idea behind Newton method is that tangent line is close to the curve and so its x intercept is close to the actual solution or the root of this function fx.

(Refer Slide Time: 07:55)



Now let us come back to our own function, we said that f was a vector psi, p, n is F 1, F 2, F 3. So there are three questions here. Then w is basically psi, p, n right? So these are the unknowns and these are the equations. So we assume that what we do

basically, instead of writing in this form, we change the form and that is called fixed point iteration.

So fixed point iteration is that this unknown w is expressed as some function which is function of these unknowns. So w means psi, p, n right? So psi, p, n is written as some function of psi, p, n something like this. Because in numerical method we actually solve the algebraic equation, right? In algebraic form it is written like this. Then what we will do, so your equation if you compare with the previous example, where we said that f x = 0.

So it becomes basically w minus Mw equal to 0. So this becomes your f the function whose root we are finding. So instead of writing fx = 0 we write w - M (w) okay, where M is function derived from this F 1, F 2 and F 3. Now we use this to estimate the next guess. So let us say you are at certain point the guess is wk. And having said this, this guess wk how to estimate the next guess.

So we will substitute here the left side we will get a new w and we call that w as w (k+1) and we of course keep on doing this. Now the criteria is that if you take, write this equation and you take the derivative. So now you take the derivative d by dw. So w is with respect to all three. So d by dw will be 1 here, right. So 1 means it is basically I, because in terms of matrix 1 is identity matrix.

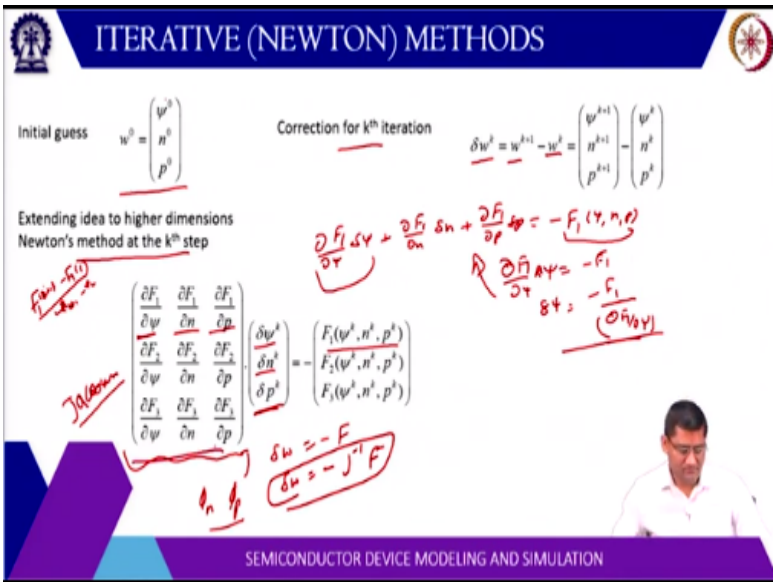
Identity matrix is matrix whose diagonal elements are 1 and non-diagonal elements are 0. Then this will be dm by d omega, so is equal to omega. Then we define some function B inverse multiplied by F. So this is how last time we defined. So M omega can be written as omega minus B inverse times F of omega. So this is the definition of basically your M omega.

Now if you take the derivative, so derivative of M omega is equal to d omega by dm that will be your identity matrix minus this B prime F plus F prime B. So B prime F plus F prime B. Now if you apply this let us say omega star is a solution, which is the actual solution. So we are starting with the assumption that there is a solution which exists for the system and that is nearby not far away from the guess.

So at omega star your F of omega should be zero. So if you substitute this F of omega star, this is omega star. So we are evaluating this as the omega star. So this is zero. So that implies that M prime is equal to I minus B inverse times F prime. So here what you have to do, we have to choose this operator B to get this expression for this M here and make sure it converges.

So one such operator that is used in classical Newton method is a derivative of F. So what you have here? M prime is equal to I minus F dash by F dash. So this will be I this will be zero. So of course, because M omega is the solution. That means M omega is equal to omega so that M prime omega has to be zero. So this is the basically theory of this Newton's method. So how it is applied let us see.

(Refer Slide Time: 12:50)



So let us say we start with initial guess. So zero is the initial guess. That is psi 0, n 0, p 0 are the initial values that we have assumed. Then for any arbitrary k-th iteration we can estimate omega k 1 s and difference of omega k plus 1 minus omega k will be delta omega k. And this is basically for the k-th step. This is the B prime right, the derivative. So we have del F by del psi, del F by del n, del F by del p.

And this derivative with respect to size multiplied by delta psi del F by del n multiplied by delta n. Del F by del p is multiplied with delta p. Now here we are using n and p. It is also possible we can use phi n and phi p. So they can also be used, but for simplicity, I have used n and p here. So that should be equal to - F (F 1).

So if you expand it here, see del F 1 by del psi times delta psi plus del F 1 by del n times delta n plus del F 1 by del p times delta p is will be equal to -F 1(psi, n, p). So that is what we are doing here. So by writing this equation, what we are trying to find out, we are trying to find out this delta psi, delta n, delta p.

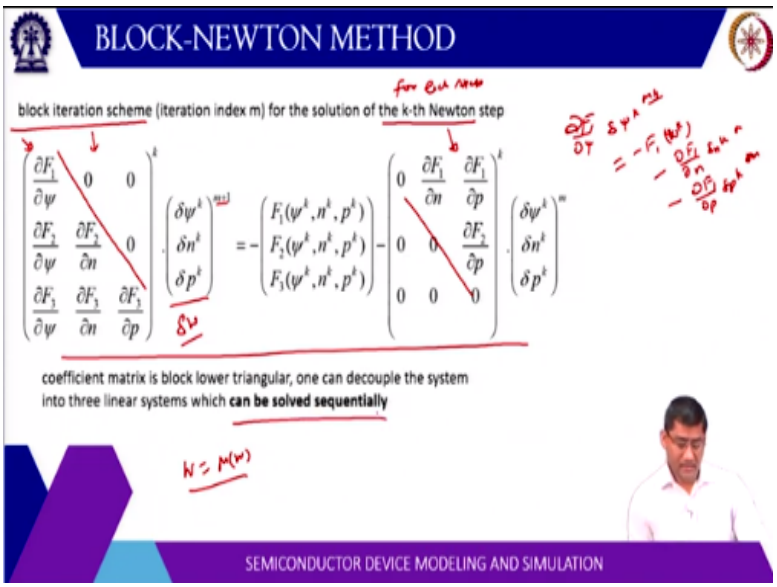
So that means if you consider only let us say one unknown, so what we are doing del F 1 by del psi times delta psi is equal to –F 1. That means delta psi is equal to –F 1 by del F 1 by del psi. So this is exactly same thing we did for one dimensional case right? So now this is expanded to three dimensional case basically. So we write this equation for all the functions F 1, F 2 and F 3.

Now this is called Jacobian. This is a derivative. Now we have algebraic equation right? So this Jacobian is also calculated from the, calculated numerically, because we do not have expression for F 1. So we cannot take the derivative using expressions, but numerically we take the derivative. So this numerical derivative you can find out this F del F by del psi will be F 1 at let us say x plus or i plus 1 minus F 1 at i divided by h x and so on.

You know in three dimension of course all the three have to be added. So that way we take the derivative. So here is the psi. So it is basically with respect to psi. So it will be psi i plus 1 minus psi i. So change in psi is with change in F. So these Jacobians are calculated at each iteration. Then this let us say this is J times del w is equal to minus let us say this is F.

So del w is basically your minus J inverse times F. This will be your solution. Now inverting the Jacobian again is computationally demanding. So what trick we can use to overcome this? Instead of writing the full Jacobian we can split this Jacobian.

(Refer Slide Time: 16:38)

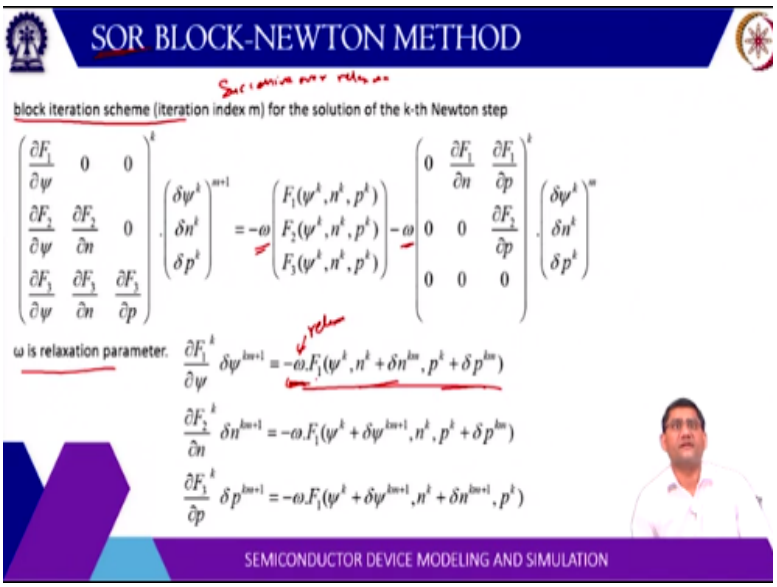


And that is called block iteration scheme. So here we have again created the same thing w is equal to M w. So you see here, this is taken, this diagonal matrix is taken. This is a lower triangular matrix. And then there is upper triangular matrix here with diagonal element zero. So if you see here they are exactly same equation. So what we have done, for these values, we have taken the previous iteration value.

And for this this is the new iteration value. So if you write this equation what you will get, del F 1 by del psi times delta psi k is equal to –F 1 psi k, let us write w k minus del F 1 by del n delta n k minus del F 1 by del p times delta p k. And now iteration number is you see it is M + 1. So this is M + 1 iteration. And these are the M-th iteration. So for each step so this means, for each step we are solving this equation iteratively.

So for each step let us say for k-th Newton state, we are again doing iteration m equal to 1, 2, 3 and so on. So for M-th iteration we use this one till we get the exact solution for delta psi, delta n, delta p or delta w okay. So this is called block iteration scheme. So here we are not inverting the matrix and solving it, rather we are splitting it into half and writing w equal to M w format and iterating it till it converges. Then this can be solved sequentially.

(Refer Slide Time: 18:52)



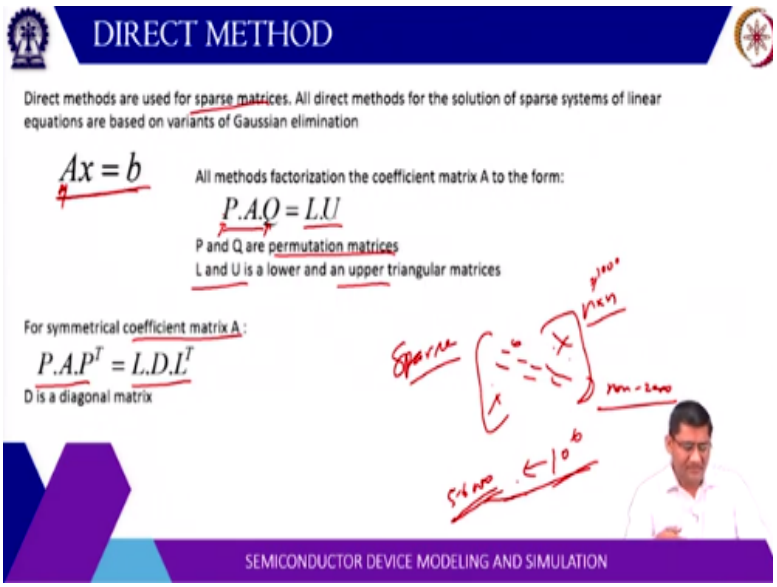
Another improvement we call it block iteration scheme with successive over relaxation, successive over relaxation. So that means, instead of subtracting these values exactly, we can take a factor of it because it may be possible that you know in some case if the function is not good enough, then the answer may be actually kind of oscillate. So it may overestimate. So it will take more time.

So what we do? We just multiply this factor by omega. So omega is a scaling factor, some number. So we instead of writing this F 1 simply we write minus omega here. Now you notice here, this term is basically if you see is what –F 1 then psi, n and p. So now we are subtracting this one del F by del n. So this can be written as minus del n and this can be written as minus del p.

So this is what is written here, del n del p here. And it is multiplied by this factor omega, which is relaxation parameter. So again it is the same iterative method, but with some relaxation parameter. Now these parameters you know people have estimated that you know which value of this parameter will be good for which scenario.

So these are basically you know subject matter of numerical techniques, but this method can be used and one can use the appropriate resource to find out what should be the appropriate value of this relaxation parameter.

(Refer Slide Time: 20:51)

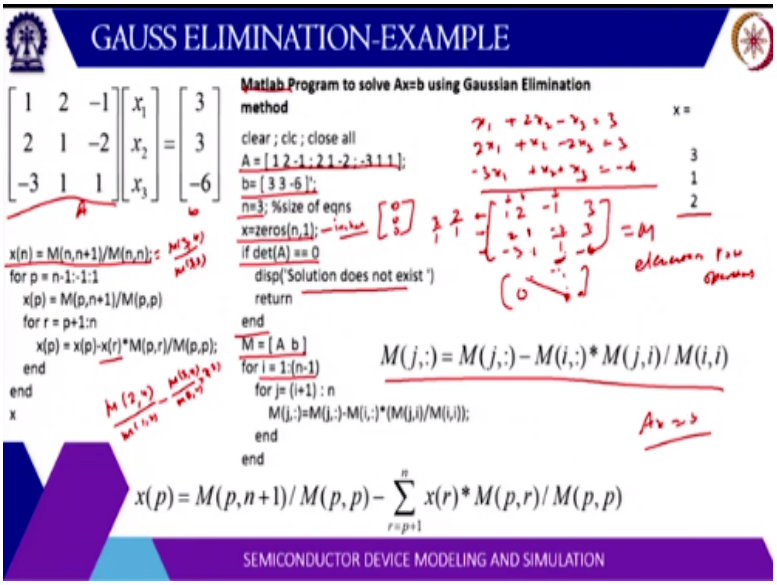


Then what we have discussed are the iterative methods. Other methods to solve the semiconductor equations are the direct method. So there we have this algebraic, now after the discretization we get algebraic equation of the form Ax = b. We can solve it iteratively or we can solve it directly. So what we do we again this convert A into upper triangular and lower triangular matrix.

So this A is written as, what we do basically, we use, we do not directly write it as A LU because some conditioning is required because if matrix A is not conditioned properly, then we take help of the other matrices, let us say p matrix and the q matrix and these are called permutation matrices. And then PAQ is written as LU decomposition. So where L is lower triangular and use upper triangular matrix.

For symmetrical, let us say matrix A is symmetrical. Then P and Q are related. So Q will be simply P transpose and U will be simply L transpose and these are diagonal matrix. So this can also be used.

(Refer Slide Time: 22:07)



Now to solve this Ax = b, we use some methods. So one method I have used here for illustration purpose so that you can see that how do we use the Gauss elimination method. So Gauss elimination method is basically you have this equation. So if you write here, this is x 1 + 2 x 2 - x 3 = 3. Then your second equation is 2 x 1 + x 2 – 2 x 3 = 3. Third is -3 x 1 + x 2 + x 3 = -6.

So these are the three equations. So what we do? So we define this matrix A. We define this matrix b, so this is A this is b. Then we also tell what is the size of the equation n = 3x. So there are three unknowns. Then we define the solution matrix. Of course x will be let us say initial guess, this is our initial guess, let us say initial value. These are the let us say zeros of length n which is 3.

So we have defined x to be 0, 0, 0. Then we determine what is the determinant of A. And if determinant of A is zero, then solution does not exist basically, where there will be infinitely many solutions. So it will just return and end the code. Otherwise, if it passes this one the determinant of A is nonzero, then we make a matrix consisting of A and b. So A and b basically it becomes 1, 2, -1, 3; 2, 1, -2, 3; -3, 1, 1 and -6.

So this is your matrix M. Now what we do, for matrix M this is what is done here. What we do, starting from second and third row we subtract. So this is the say first element. So we multiply this whole row by first element of second row that is 2 and divide by the first element of the first row. So these coefficients become equal and then of course we substitute, subtract it.

Similarly, for the third one, we multiply this by 3 divided by 1 and then subtract it so that these elements become zero. So these are the elementary row operations, these are called elementary row operations. So in terms of for i equal to 1 to n - 1 okay. So 1 to n - 1, j is equal to i plus 1 to n. So initially let us say i is 1. So we will do j from 2 and 3. And this is written here.

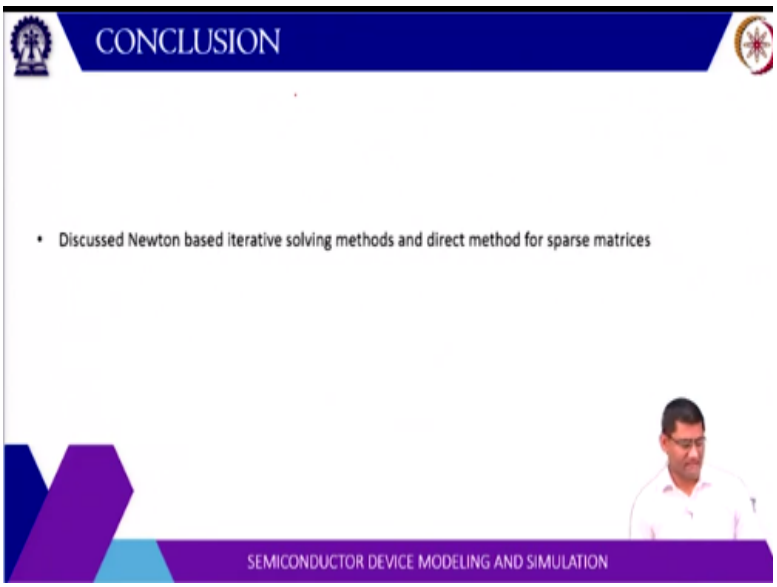
That means, for each row we are making it the first element zero basically. So this i means first column, i = 1 is first column. So we are making the first column of second row, third row equal to zero. Then i = 2. Then we are making the second and third row the second element zero. Then third one we do not have to do because this is up to n - 1 only, because this one element has to be left, right?

So what it will do basically, it will convert this into a upper triangular matrix. So these will be non-zeros and these will be zero. So now we can directly tell does this some element here times x equal to -6. So that will give you the x 3, right? So this is the solution that is done here. x n is defined as M n of n plus 1. So this is basically M of 3, 4 divided by M of 3, 3.

So 3, 4 means this element divided by this element. Here you get x n. Then for other n - 1 what we do? Let us say for next element M (2, 4) divided by M (2, 2) right? So because here this we have to determine. So divide by this and minus this element right multiplied by the known value of x 3. So - M (3, 4) divided by M (2, 2) into x 3 right? This is xr basically, p plus 1 to n.

So this is basically procedure is followed to calculate the values and this is the output here 3, 1, 2. If you substitute here, so if you multiply Ax you will get actually b. You can verify it. So this is one example how you can code this process in mathematical format in some code. This is MATLAB code basically.

(Refer Slide Time: 27:21)



So in this lecture we have discussed Newton based iterative solving method and the direct method for sparse matrix. Okay. These techniques are used for this direct methods. They are used for sparse matrices. And sparse matrices are those where nonzero elements are actually very few, so diagonal and some nearby elements are nonzero. And most of other elements are actually zero.

So these are sparse matrices. Now what is the advantage of the sparse matrix that let us say it is some n by n right? So n maybe some 1000 by 1000. So if let us say n is equal to 1000, then total number of elements of the matrix are 10 raised to power 6. But instead of using 10 raised to power 6, we only store the nonzero element. So let us say 3 or 5 diagonals are nonzero.

So you will have some 5 or 6000 nonzero entries. Now compare this thing. 1 million entries versus 5 to 6000 entries, okay? There will be some element here and there which are nonzero. So maximum 10,000, but that number is quite less. So a sparse matrix techniques are used and these are the actually sparse matrices we get in when we discretize the semiconductor equations, because when you discretize a derivative or double derivative, you get i, i + 1, i – 1.

Rest are zero with respect to that unknown. So we do get the sparse matrices and the direct method with some value decomposition or with some, you know this Gauss elimination method and they can be solved iteratively basically. So thank you very much.