Foundation of scientific Computing

Prof. T. K. Sengupta

Department of Aerospace Engineering

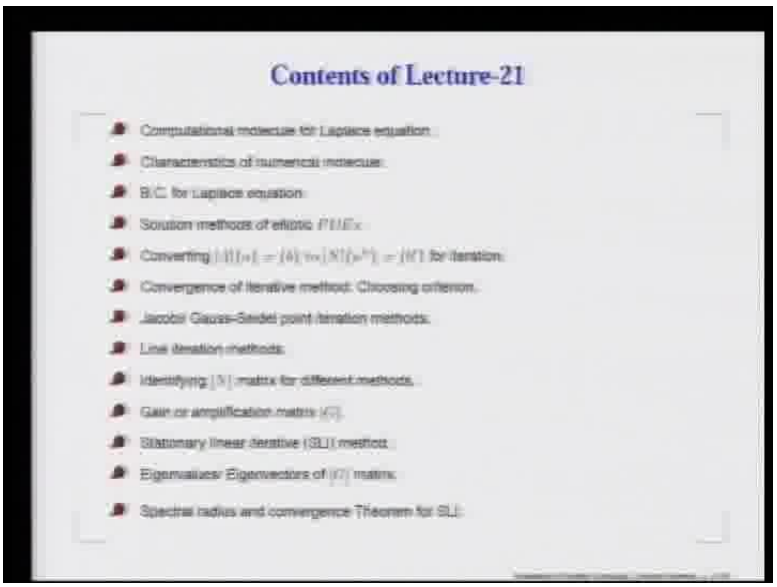
Indian Institute of Technology Kanpur

Module No. # 01

Lecture No. # 21

We presume our discussion that we are going to solve the elliptic PDE and I am going to start all over again - I am restarting. We will continue our discussion on elliptic partial differential equations and the solving methods in this lecture 21. As an example, we take up the Laplace's equation and as a brief recap we talk about its characteristics and the boundary condition requirements for this Laplace's equation.

(Refer Slide Time: 00:17)



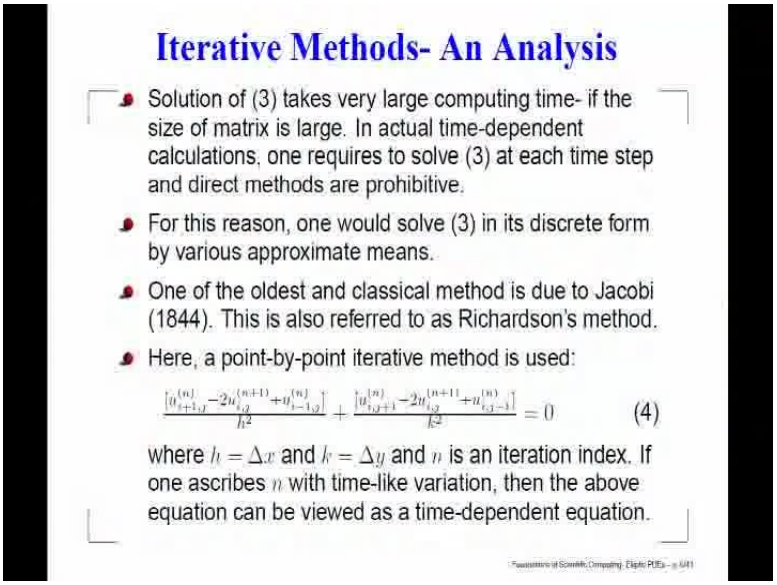
In the process, we start talking about general methodologies for solving PDEs. What we notice that we cannot invert the A matrix, so that is why what we do is, we adopt iterative methods and we replace the A matrix by an N matrix and correspondingly, the right hand changes and this is basically the essence of all iterative methods.

Therefore, we are interested in knowing the convergence rate of these iterative methods. There are various criteria that have been talked about. How we choose this criteria to find out whether we have converged or not?

Historically, this problem goes back to the methods pioneered by Jacobi, which was also done later by Richardson. It was improved simultaneously by Gauss and Seidel; these are all point iterative methods. Having discussed about these point iterative methods, we switch over to the line iteration methods and one way of analyzing any of this method is trying to find out the equivalent N matrix that we have talked about earlier on for these different methods. Since this has become now a pseudo time integration, we can talk about a amplification matrix and when this amplification matrix does not depend on the iteration index, we call that as a stationary linear iteration. We spend most of our time talking about this stationary linear iterative method or SLI methods and we talk about relating this SLI methods convergence weight with Eigen values and Eigen vectors of the G matrix.

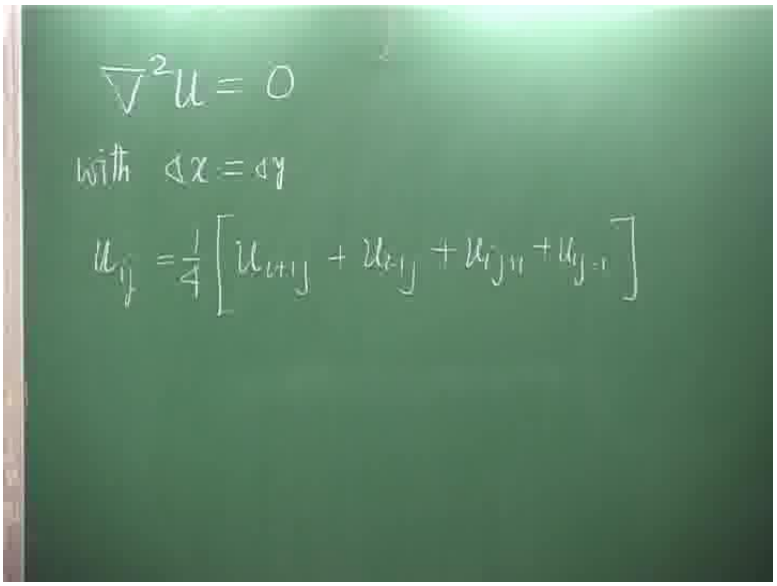
I think in this context what is important is, the maximum Eigen values of this G matrix which you call as the spectral radius. We enunciate various convergence theorems for stationary linear iterative methods discuss their amplifications.

(Refer Slide Time: 03:38)



See basically, we are talking about solution of elliptic equation and the simplest of all of them is the Laplace's equation. We are looking at this and if I take same spacing that is delta x equal to delta y then of course, that means h and k will be same and we get the discrete equation let me known like this; this is one fourth of the neighbors as shown in slide.

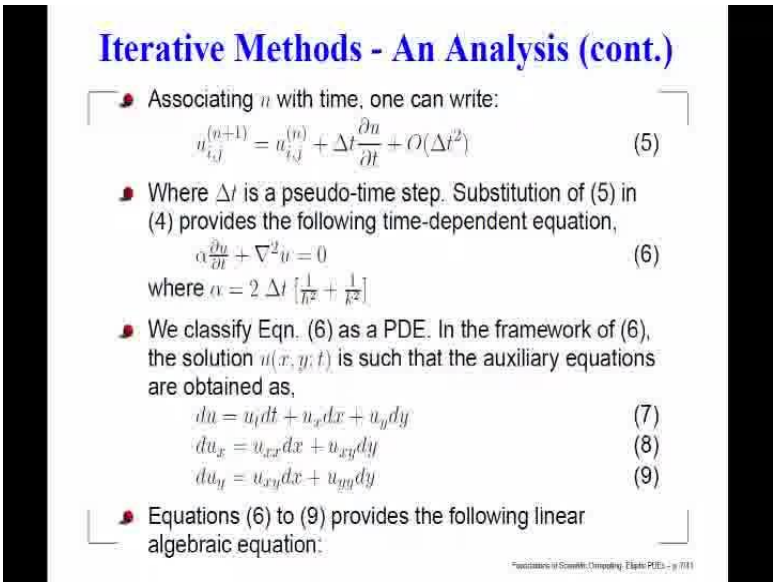
(Refer Slide Time: 04:02)



From there, we actually commented upon the maximum principle for this solution. We noted that if you are looking for either the maximum or the minimum, it has to occur on the boundary; it cannot occur in the interior because, interior points are nothing but averages of the neighbors. So, average cannot be greater than the constituents. Of course, it is easy to conclude from Laplace's equation, the maximum or the minimum will reside on the boundary.

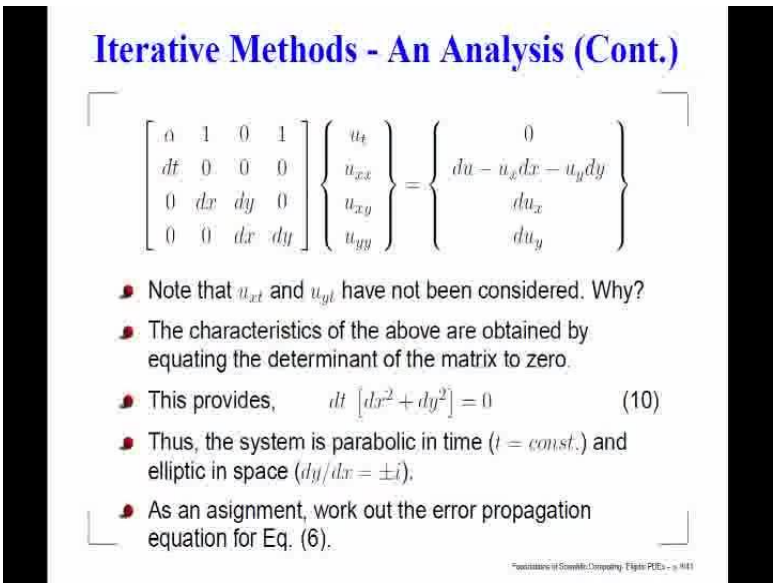
However, it is not the case that you should be generalizing for Poisson equation, where you have a nontrivial right hand side. We commented upon the fact that direct solution of linear algebraic equation like this takes enormous effort. The matrix size of N by N it amounts to NQ operation, whereas one could actually go about to iterative methods and hope for lesser amount of work.

(Refer Slide Time: 05:52)



The oldest such effort is due to Jacobi and which was subsequently adopted and studied extensively by Richardson, so it is also called the Richardson method. Here, the point by point iteration is attempted by looking at the current iterate for i j th node in terms of the four neighbors.

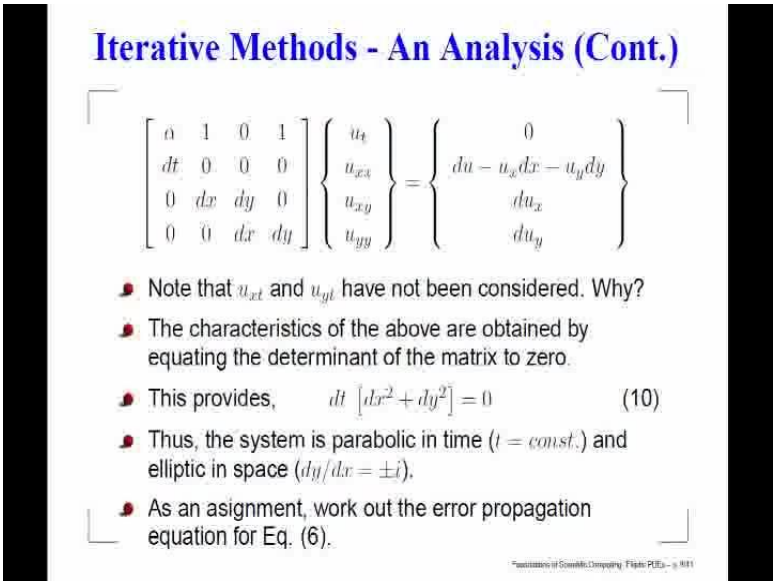
(Refer Slide Time: 06:06)



We have actually done this path before, so we just note that iterative method is equivalent to bringing in some kind of a pseudo time derivatives, so that renders the problem a parabolic flavor in time because the characteristics are now given in terms of this complex conjugate in

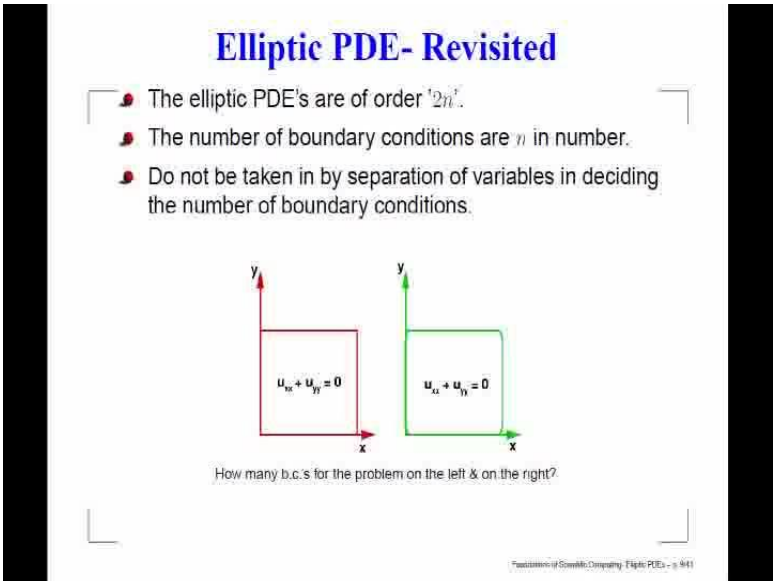
space that implies that it is a elliptic in space and of course, t equal to constant implies this parabolic in time.

(Refer Slide Time: 06:40)



As we have seen this equation that we have written here, the equivalent time dependent equation, we have written it down. So, we could actually also try to write down an error propagation equation. So, if I make some error at t equal to 0, how that error propagates? You can follow the methodology that we have seen in couple of lectures ago for 1D convection equation. You can follow that and you can try this, what we have written down there.

(Refer Slide Time: 07:05)



Now, this is one thing that is of paramount importance is that elliptic PDEs are always even order because, the characteristics appear as complex conjugate. So, you do have to have an order to be in 2n in space and this is something which might some of you may find troubling in the number of boundary conditions. I am not going to go through the theoretical aspect of it but, it is just to note that if you have order 2n you require n boundary conditions. This confusion actually arises due to the way we often approach this problem via separation of variable.

For example, if I like to solve this Laplace's equation in this rectangle or the square box. What we usually do that we separate the equation as a product of a function of x and a function of y and then, we see for solving the x dependent path. We need to give boundary conditions on this edge as well as this right edge.

We actually say that we need two boundary conditions in x and similarly, we also need two boundary conditions in y that is along the x-axis and along the top edge. Now that is what you would be implying to respond back saying that we need four boundary conditions. Unfortunately that is not the case, the twist in tail comes actually, if you look this corresponding figure here; it is almost identical what I have done is, just simply round the corners.

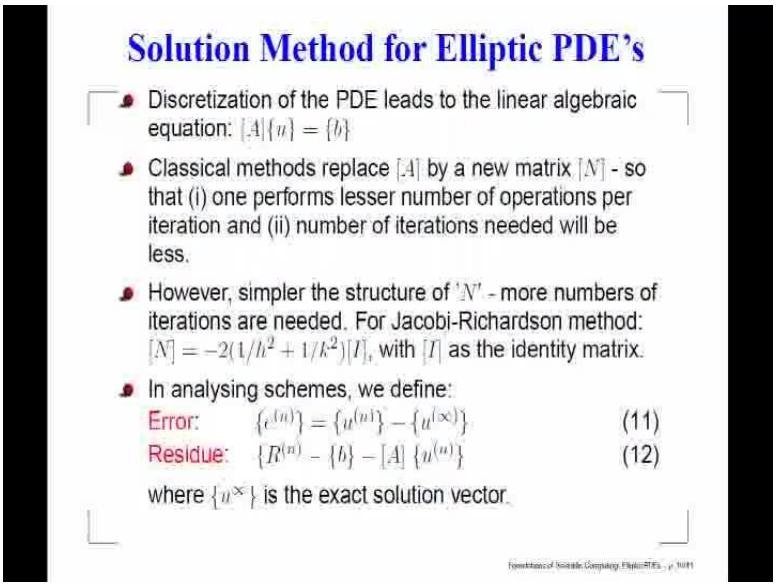
Now, what do you do? The first thing we notice that separation of variable is no more possible because you cannot split it cleanly the x dependence and y dependence in terms of the boundary condition. Whenever you do separation of variable; you do actually have to keep in mind that it is not only the differential equation that you are separating; it is also how cleanly you can separate the boundary condition. It so happens that because of this fill at the cornered edges - rounded corners - of this domain tells you that you cannot do it.

What you require is basically, one condition on all the four edges of a single boundary. So, think of this whole edge to be part of a single boundary and that is what you see. Let me just give you a brief history of this Laplace's equation. During second world war towards the end, it was Robert wiener at MIT; he actually send a proposal to US government saying that we should be able to solve flow past aircraft by solving Laplace's equation. So, it was as simple as that even then, people had to realize the importance of this equation and it was not that one could solve it analytically. So, he suggested a numerical solution even in those edges where you did not have.

Well, you just have seen the appearance of digital computer towards the end of the second world war but, he was anticipating things to come and he made that proposal to US government; the war got over, the proposal was shelled. However, another two decades downstream the scientist that McDonnell Douglas and Boeing, they got around solving the same problems.

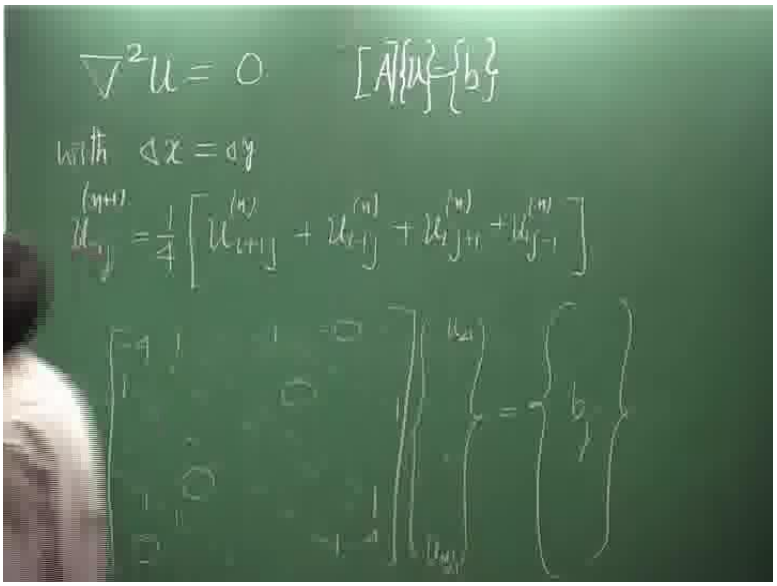
You can see that solving a Laplace's equation over a complicated shape is something that was considered very vital important for technological development and it requires actually numerical solution, you do not have any other way of solving it and that method is called the panel method. So, a subject called panel method totally depends on solving Laplace's equation over complex geometry. So, that is why it is so important that we understand the importance of boundary conditions and we got to apply with rather carefully.

(Refer Slide Time: 11:40)



Now, coming back to the actual methodology in solving the linear algebraic equation, let us say A u equal to b. What we do in the classical methods like we talked about Jacobi or Richardson method or the subsequently, it was attempted by Gauss and Seidel and what in those method one attempts is basically instead of solving this equation A; we replace this A by a new matrix N.

(Refer Slide Time: 12:24)

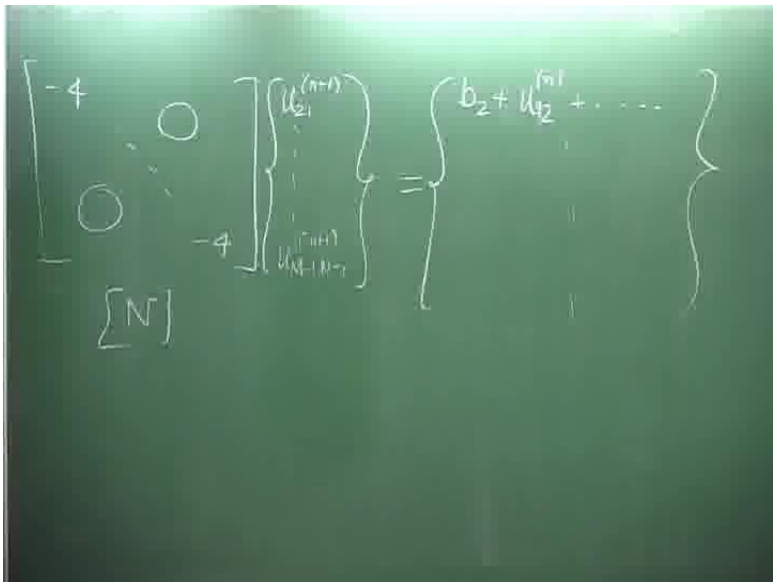


How does it happen? We have seen that if I take delta x equal to delta y the A matrix had this structure. We have 4 along the diagonal and we have 1 along the super diagonal, 1 along the sub diagonal and then, 1 along the pitch N away and another diagonal row of 1 the line which is actually one line above the point in question.

We do not actually solve it in this way in Jacobi method when we actually ascribe a superscript here. What we are actually doing? You can notice that we are in a sense recasting the original problem which was Au equal to b. When I write the original problem here, when I write all this; I have all those stacks of u's coming in. So, if I talk about Dirichlet boundary condition, it would be something like this m minus 1 n minus 1 kind of an array.

So, this is what we would be writing A u would be equal to b, so this was our problem - original problem - but when I am writing like this what I have done here as you can notice that I have not tried to solve this equation but, I have changed it in what way? We keep this on the current level.

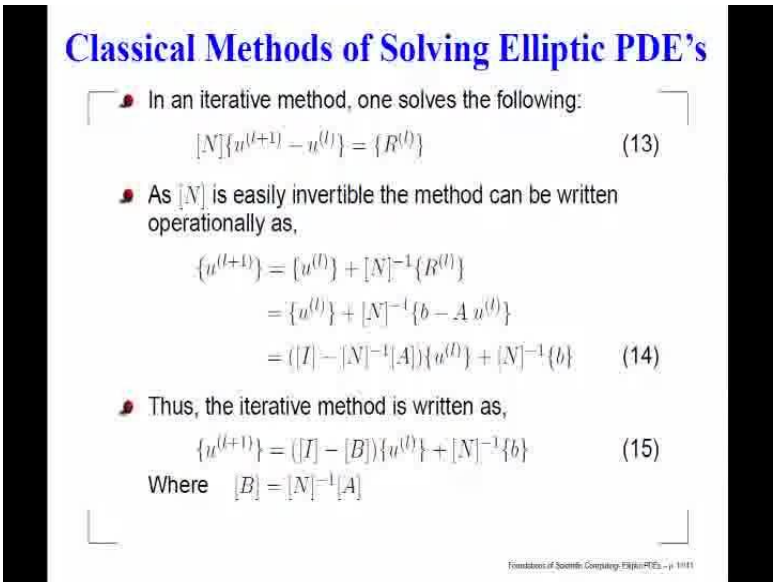
(Refer Slide Time: 14:40)



What we have done is we have actually converted this problem like this. We have transported all the quantities that we have on the other side, so whatever we had here, let me call that as b 2 and then, I have taken all of these quantity at the nth level they have been transported to the right hand side. Basically, we have really done this, so if that is there, I will just simply write u 12. So, these are at n plus 1th level all of these and these 1's I am writing at n th level.

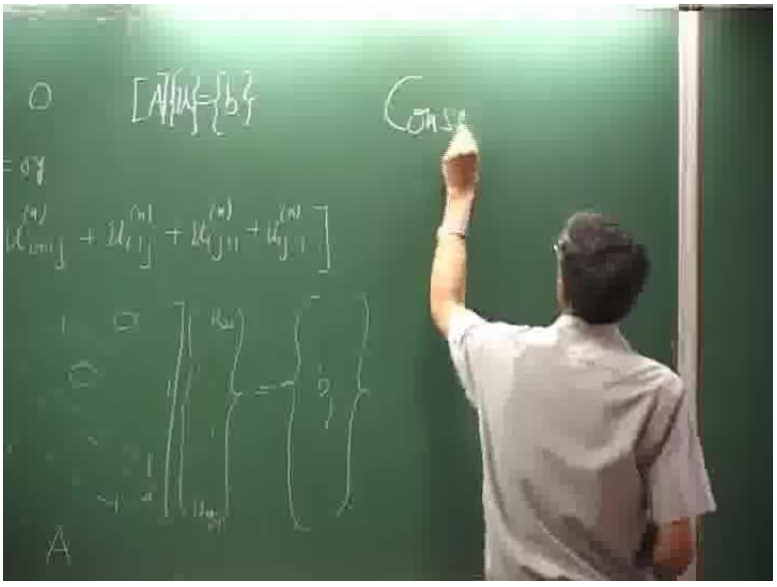
So I will write this, that is what we have done isn't it, that is what it means. What actually happened here that instead of this matrix A here in the Jacobi method, what we have done? We have just changed it to a N matrix I mean, in this case the N matrix happens to be the diagonal element of the original A matrix that is how we have done.

(Refer Slide Time: 16:10)



This is what has been said here that we actually replace the A matrix by a new matrix here. Why did we do that because we commented upon the fact that A inverse is difficult to perform but here, if I have this matrix what we have done now, we have written down this as some, I call that as a b prime.

(Refer Slide Time: 16:30)

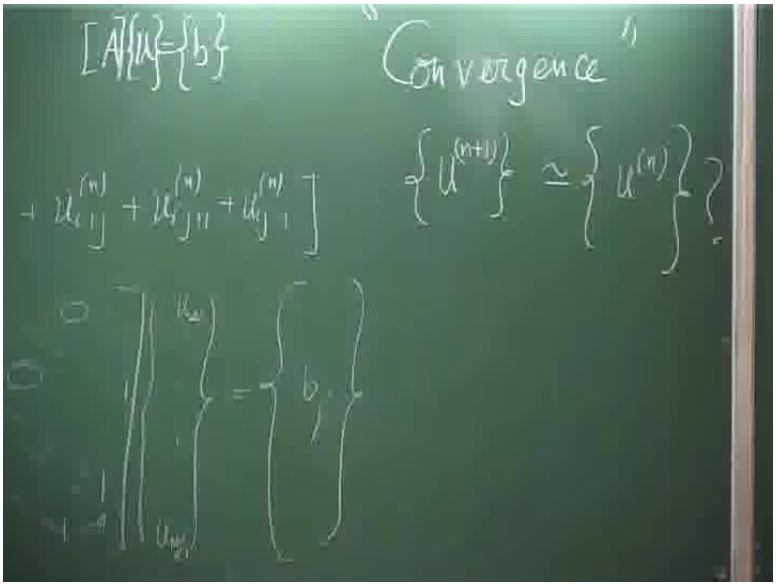


We have exchanged A u equal to b to N u n plus 1 equal to some b prime. Now, the whole idea is this n plus 1, sorry, N matrix is easy to invert that is exactly what you are doing. This step is equivalent to inverting the N matrix because, it is a diagonal matrix. All I have to do is

take A inverse of the diagonal entry, so that is what we have done. So, this is the essential idea in all these iterative methods that you actually revert to a new matrix N from the original A matrix such that N is easy to invert.

However, we have to keep in mind that this may lead to the following observation that you may end up doing fewer operations per step because, instead of n cube you will be doing something in each iteration; you will be performing lesser operation, it is just simply that four addition and one division that is what you are doing.

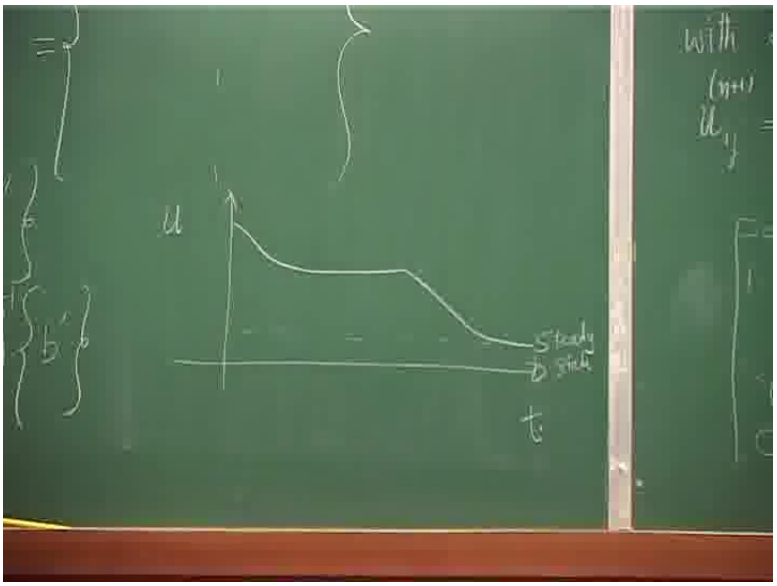
(Refer Slide Time: 18:35)



In addition, you will also like to have this number of iterations required because, we have to keep doing this; we have to keep doing this again and again. If we keep doing it where do we stop? Well, we will have to stop when we have converged. Now, what is the meaning of convergence? That is something, we will very seriously discuss what exactly mean by convergence.

You recall that we just now talk even today that when we solve it in iterative manner what we are essentially doing; we are actually solving a pseudo time dependent equation; with the hope that as time progresses the time rate will be 0 and then, that part would be gone. Convergence in that sense means that when you actually reach the steady state of the pseudo time dependent equation, so that is what we are talking about the convergence.

(Refer Slide Time: 19:48)

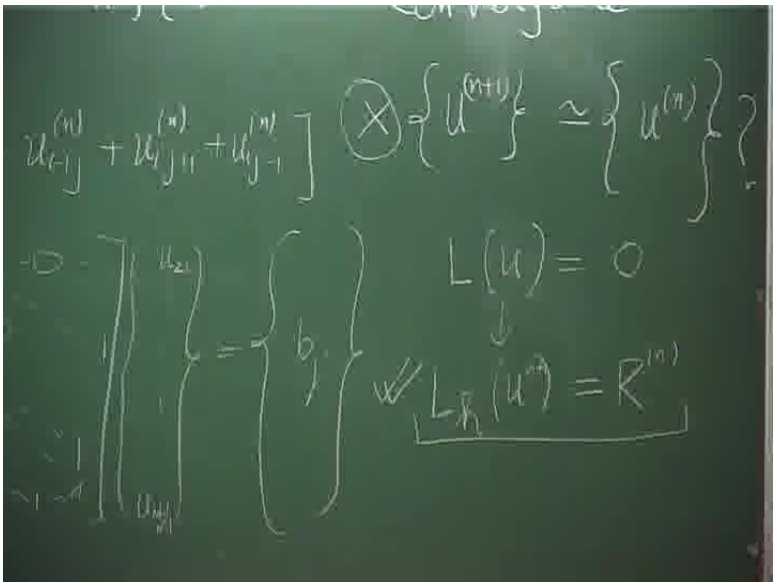


Does it actually mean that this is almost equal to this (Refer Slide Time: 19:22); we will be all tempted to say that because that is what I just now said, the time derivative will go to 0 but then, think of a time dependent problem; if I want to plot u verses t and I am looking for some steady state let us say, I know the solution, this is where eventually it should stay.

Now, it can so happen in the process of iteration that it will decay and then, it will remain straight. If I am not careful; I may actually declare my convergence has occurred here. Then what happens is, suppose, we just simply continue further then again, we may see that it may actually take much longer to do that and this is exactly what happens in computing.

What happens is, we are not going to adopt this; this is a wrong way of looking at convergence. We can always have this false perception of steady state having arrived, which will not happen.

(Refer Slide Time: 21:02)



What exactly we all looking for them. Look at the following problem that we have some l let me write linear equation that we are solving, so this is our problem statement, this is a continuum problem. Now what we do is, we actually have a discrete equivalents of it and that is what we solve, so I will call that as L of h because, I have chosen grid size which is quoted here as h, so I am actually trying to get this.

Now over and above what I am doing? I am iteratively doing this and if I do that does it become 0. Well, the answer is no because, what happens is our solution is not the correct one. So, if I plug it in there it will not do that. I will get something what I will call it as residue that is what I have written here. Equation 12 tells you that if your iterate is not a good guess then of course, this one the discrete operation would not be giving you 0, if it does then you have already arrived at this.

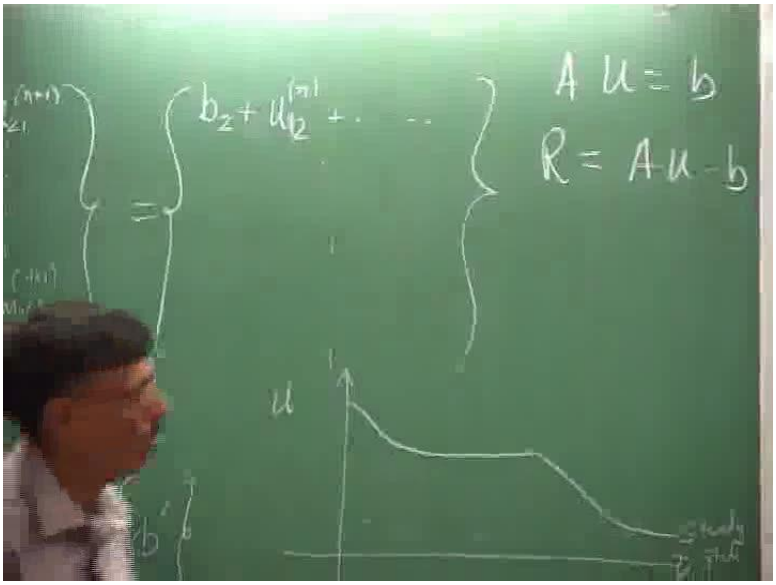
So this is what you should be checking at, not whether the solution has failed to improve; failure to improve with iteration could be a bad attribute of the method chosen whereas, this satisfaction that in the discrete sense the solution actually satisfies the governing equation is what you are looking for, that is what we are computing. We are computing in the sense that in the discrete sense that operations would reach to the satisfaction. So, when this R n goes to 0 the residue goes to 0 that is where we have convergence.

We need to look at that. So, how do you do that? Of course, I can subtract one from the other and would give me this thing. So at every time step - I mean every iteration step - we should

calculate this L h of this and check if that has gone to our desirable level of convergence or not. Please do understand that it would never be able to take you to absolute 0 at every node. It will be dependent upon machine precision it will depend on method chosen.

What you like to prescribe affront is that ok I am looking for 7 digit accuracy or 10 digit accuracy so I will set this equal to I will check, if this is equal to some epsilon; if this epsilon I may decide 10 to the power minus n; n could be 6, n could be 10 and n could be 14 but of course, you should never attempted beyond the machine precision depending on how you describe the real numbers.

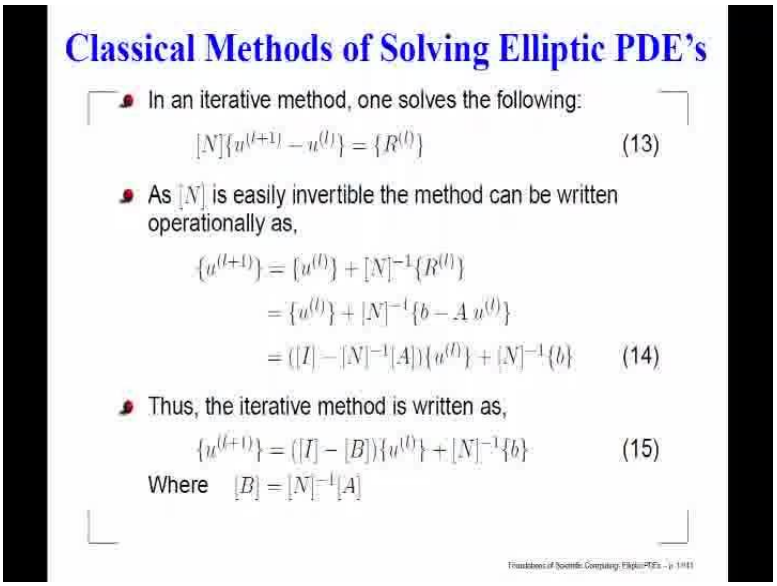
(Refer Slide Time: 24:47)



((Student talks)) Yes ((Student talks))

Yes, I think there is a bracket missing here (Refer Slide Time: 24:33) R n or there should be equal to here. The mistake I will have to take care of it. So that is precisely what we have talking about, we are trying to solve A u equal to b, so residue is defined as A u minus b. Thanks for spotting that.

(Refer Slide Time: 25:37)



You see this is one thing that we have to keep in mind that whenever we take the iterative method; we first of all change A to N and then, we look at the solution residue or discrete difference equation not the u n minus u n minus 1that is not what we are looking for; we are looking for in the discrete sense if this R is uniformly going to 0 up to some tolerance or not.

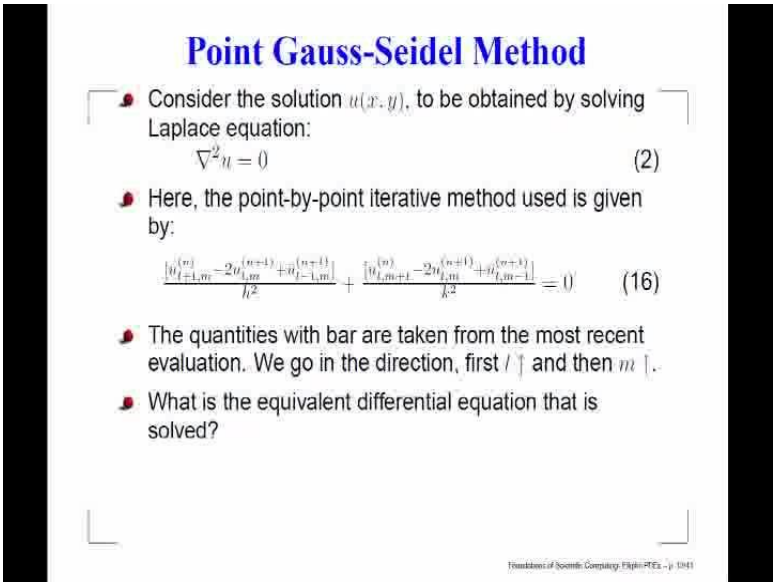
Basically, then we actually solve an equation of this kind. I write that altered new metric N operating over the correction that I want to perform at that level and that is determined by the residue calculated over the previous step, so that is what we like to do. Basically, then what happens is that u l plus 1 could be u l plus N inverse R and R itself is b minus A u l and so, we get this. You can see that in a sense solving 13 is equivalent to improve the solution by this algorithm, where B matrix is nothing but N inverse A.

What we need to do then, I think I have jumped something just make few more observation. You can very clearly see that if N is identically equal to A then, what happens? B is identity matrix and then, what happens to this part I minus I and then, this is A inverse B. So, A inverse B is the original solution that is how what we wanted to do u should be equal to A inverse B. So, what you notice that success of your method will depend on how closely this B matrix is to the identity matrix that means, how close your N matrix is to the A matrix. Here, we have a sort of a ((dicortiunea)) we want to take N. so that it is easily invertible, at the same time we are demanding N inverse A should be equal to identity matrix.

This is a sort of a conflict and that is what we notice that we picked up the A matrix and the Jacobi method what we did the N matrix was nothing but the diagonal entries only.

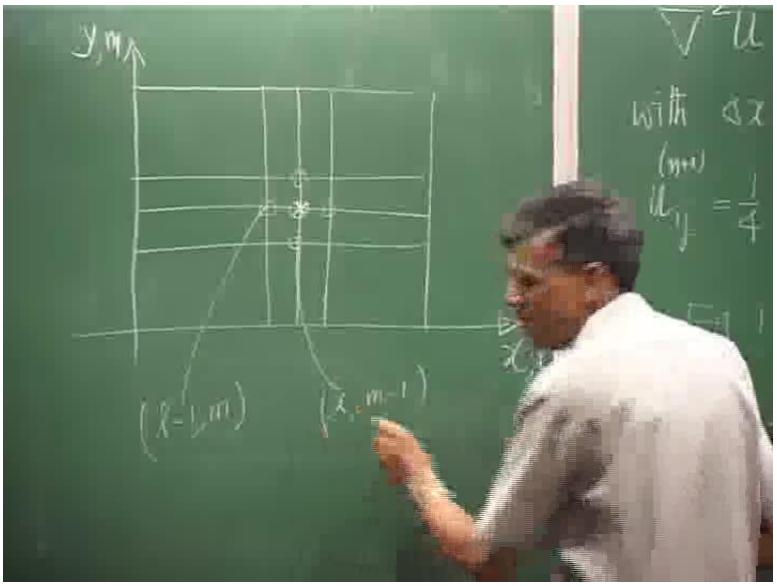
Of course, they are not same; of course, N inverse A will not be equal to identity matrix it will be quite different and that would make this process slow to conduct. So, what we are essentially saying that if you could have performed the magic or picking up N such that N inverse A is identity matrix that is equivalent to your direct solution, it is no more an iterative solution; you just got it in one stoke that is the solution N inverse B.

(Refer Slide Time: 29:06)



Basically, it is a sort of a demand which we are not going to meet that we cannot really have it in e t 2; we cannot have N equal to A and then, say it is an iterative method e t 2 is not going to be. Now, in that process, what one can do is improve up on the Jacobi method and that is what was suggested by Gauss and Seidel. Here, what you notice what has been done this is something that would make common sense to all of us if we notice what we are doing.

(Refer Slide Time: 29:45)



Now, if I take a 2D problem on the one side I denote it by x co-ordinate or the index l and on the side way it indicate y or the index m then, we are solving the problem in a domain like this then you have noticed that in solving the Laplace's equation or the Poisson equation; we have a 5 point molecule. So, if I am doing this I am involving these 5 points then iteratively solving it.

In the Jacobi method what we did? We kept this central node at the current level while the neighbors who are all at the predecessor level that is what we did. Now, Gauss and Seidel made this observation that if I am working here (Refer Slide Time: 30:38) that means what? At this level, I have already got this solution that is already available and so why is this? Because, we have gone like this and we gone up. So this point we have making here that when we reached here, we already have this and this at the current level and so, we should use that information. If you use that it is a kind of a hopeful session to begin with that this will be a better way of solving the method than the Jacobi method and that is precisely what we have written in equation 16 here.

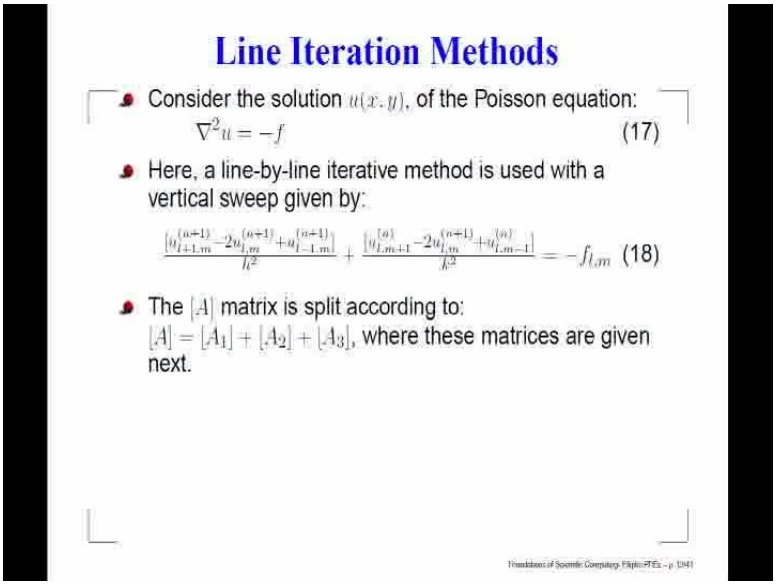
Look at this we are at the l m th node, so that is why we are indicated it by superscript n plus 1but look at l minus 1 m; l minus 1 m is this point, so this is your l minus 1 m, so that point is already available at the current level and that is why we have indicate it at n plus 1 but, please do understand that although we are writing the same superscript n plus 1 there is a difference between u l n and u l minus 1m because u l m we are trying to solve whereas, u l minus 1m

we already have the solution, so that is why I have indicated it with it over bar. This will appreciate as we go along.

Now the same way if I look at l m minus 1 this point (Refer Slide Time: 32:14), so this is l m minus 1; this is also available at the as a new estimate so that is what we have done here in the last entry that we have indicated again by superscript n plus 1 but, to distinguish it from u l m we have put a bar over it and that is what we have say that we have the direction first l increasing from left to right and then, we have gone from bottom to top m increasing.

Now, the same question that we asked before we can ask the same thing here. In doing this we have probably given up on the actual equation that we wanted to solve that is a del square equal to 0.

(Refer Slide Time: 33:18)



What is the equivalent differential equation that we have to solve? We can think of that so what we have done? What would be our instinctive guess? The Gauss-Seidel method would be better than Jacobi method or what of course, it is a loaded question answer is it better otherwise, why should you even try or it is a bad method. We should be knowing it shortly but the point is what we are trying to do is, we are trying to gainfully use the information already gone out what we have already acquired.

So, if I have this solution at l minus 1 m and l n minus 1 and I am going to use it; so if I am going to use that so the hope is that it is going to improve my convergence rate that is the

whole idea. To understand that let me look at some alternative method and in the combined fashion will analyze the whole thing together. Here, so far what you had talked about in the Jacobi method or Gauss-Seidel method; we are evaluating the quantities point by point. We are taking one point at a time and we are using this algorithm or the previous pages algorithm and we are doing that.

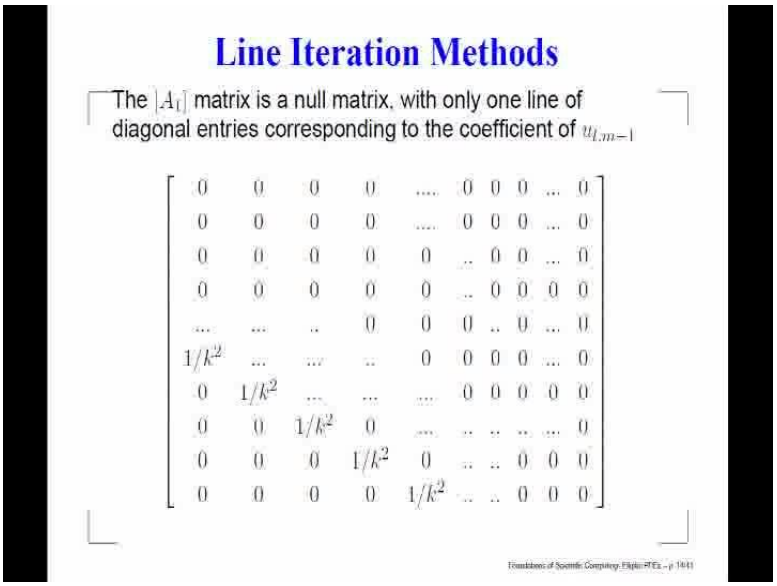
The next logical thing for us would be instead do better by solving this equations line by line and that is what we are looking at; we are talking about here line iteration method. So, what we are talking about that if I am at this level n then, what I could do is? I could write down this discrete equation where we write down all the quantities along that same m at the same level.

Please do understand here that all these quantities are written with n plus 1 and they are all treated together, that is why they are no over bars here. So, all these quantities are the unknowns whereas, if you look at this one what we have done? We have kept it at the old level but although, if I am doing line by line; so if I am working on this line already this line is available to us.

So, this line iteration method that we write it down is a kind of a variant of the Jacobi method, so I will call this as line Jacobi method because, I am solving it line by line but, I am not using the most recently available information. If I would have done, I should have change this but I have kept it at the old level itself.

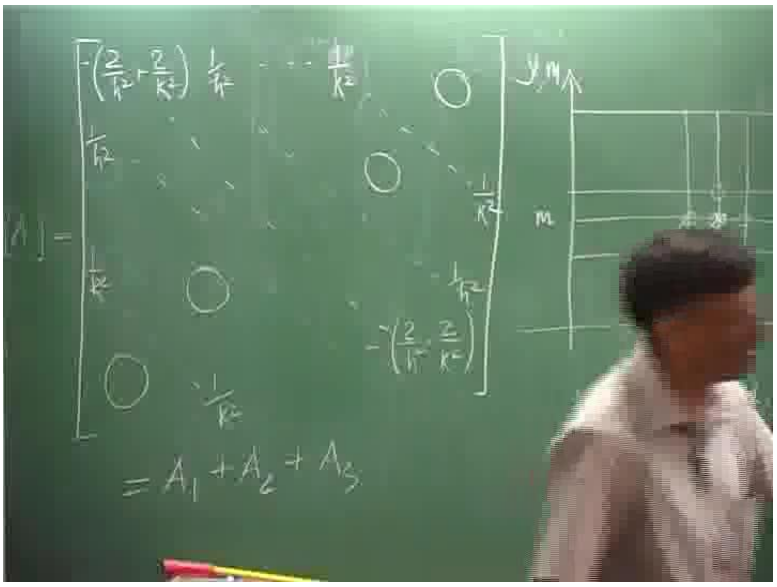
What happens is, we have this discrete equation and we write it for all the nodes and they are all coupled equations and you can see why we spend so much of time talking about solving tridiagonal equation because, you can very clearly see that if I put all the n plus 1 term on one side and n terms on the right hand side then, we are going to get a linear algebraic equation with tridiagonal matrix entry.

(Refer Slide Time: 37:04)



So that is what we are doing. To understand that let us split this composite A matrix into 3 parts, which I have called here as A 1, A 2 and A 3 and let us define them as we can see. Now, this requires a little bit of careful observation, so let us do it carefully.

(Refer Slide Time: 37:40)



So, if I look at my A matrix; the A matrix that we have written here is nothing but, I have collected 2 by h square and 2 by k square this is along the diagonal, so this is nothing but the coefficient of u l n that is your diagonal entry.

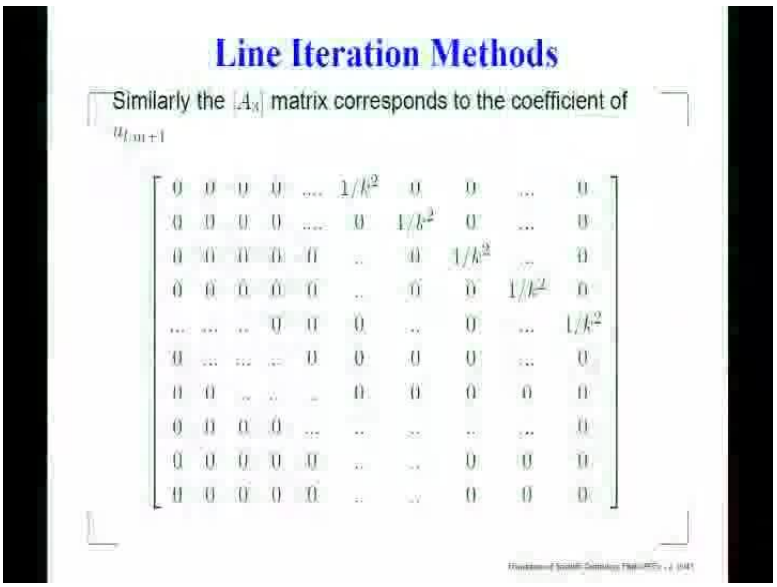
Now, if I look at this equation that l plus 1 m has an entry 1 over h square. So, here I will have 1 over h square that corresponds to u l plus 1 m, u l minus 1 m also would have 1 upon h square, you can see that; that is what we have done so, u l m has minus 2 by x square from here and minus 2 by k square that is what we have written.

Now, look at you have m plus 1 that would be so this will be 1 upon k square that would be written here, so this pitch is that number points in the l direction because you have gone to the next level. So, you are actually looking at the coefficient of this point (Refer Slide Time: 20:19).

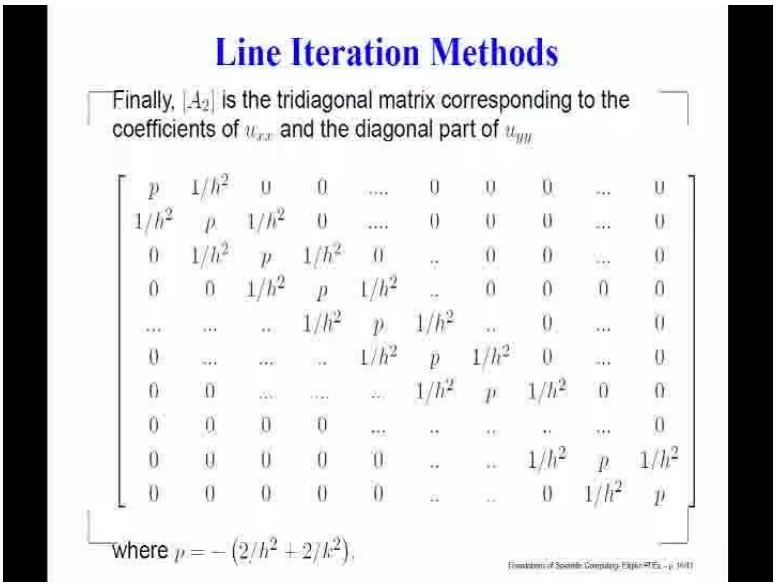
Now the same way, I would also have coefficient of this point that will appear here. So, what happens here? I have three continuous diagonal entries and this is what I have. Once again, we have this parts matrix with this kind of banded structure and what we are doing? We are writing this A in terms of three sub matrices whose summation is the full matrix A.

So that is what we have shown. What we have written here, this is A1 which corresponds to the entries of u l m minus 1 and that you can see 1 by k square so this part (Refer Slide Time: 40:03) rest of it is all 0 this is this; so this is our A1. Similarly, I could also take this one 1 over k square, I will put everything equal to 0 that is my A 3 matrix and what remains of course, is the tridiagonal matrix that we have here. So, this is what we call as the A2 matrix.

(Refer Slide Time: 40:14)

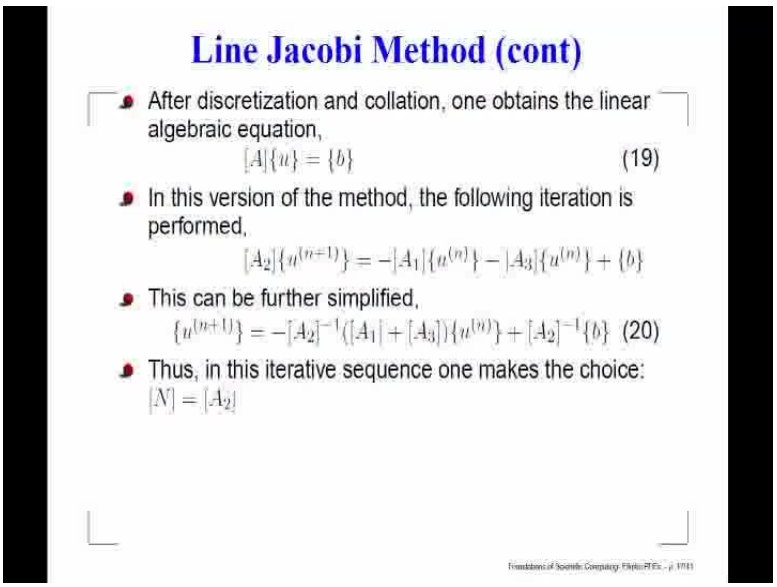


(Refer Slide Time: 40:27)



What does A2 correspond to? A2 correspond to the coefficient of u xx plus the diagonal part that minus 2 by k square part is also included here that is what we have written here. So, this minus 2 by k square comes from u yy whereas, rest of it all comes from u xx and that is what we are doing in this particular set up.

(Refer Slide Time: 41:13)

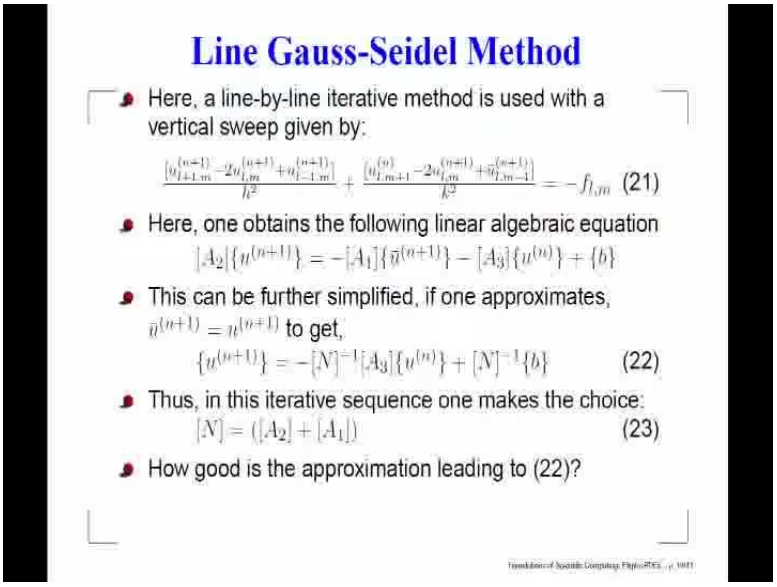


Now, what you have seen that linear algebraic equation A u equal to b has been written in this format in line Jacobi method. What we have done here, we have written A2 multiplying on the current iterate this is tridiagonal system and everything else is put on the right hand

side. Of course, we already know the tridiagonal matrix can be inverted with linear proportional operation, we have talked about that it takes about 5 to 7 n operations and then, we can follow this methodology.

In this particular choice, what have we done? N matrix is equal to A2. So, this is an improvement over the point Jacobi method; point Jacobi method only had this as the entry now this new matrix is nothing but this tridiagonal band. We expect it to be better because, this N matrix is closer to A then, the previous N matrix.

(Refer Slide Time: 42:40)



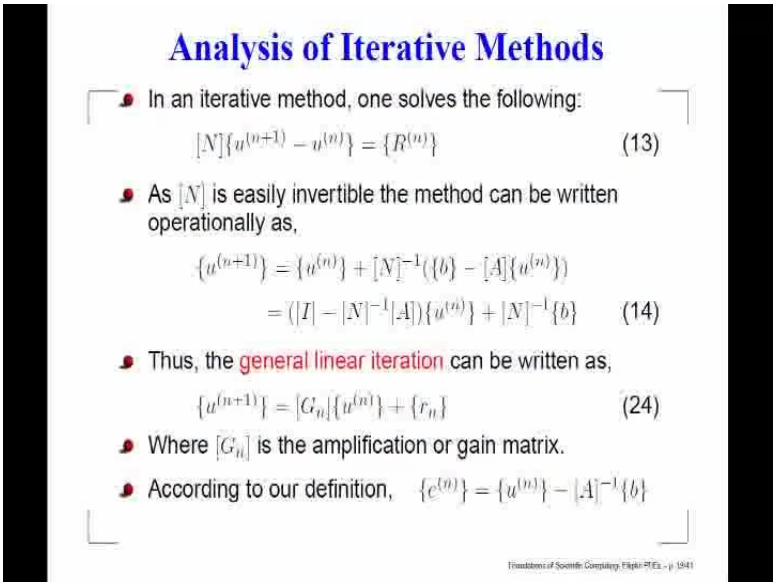
The previous N matrix only had the diagonal entry matching with A but now, three diagonals are matching with A, so this have to be a better approximation. So, that is what we expect from this line Jacobi method, so we have now made some observation. Now, as I told you that when we are working on the m th line this point was available to us. We did not use it in line Jacobi method, in line Gauss-Seidel method you just simply use that also.

Basically, we have put in the last entry here and please do understand that this is with a bar indicating that we take it from the most recent estimate of the solution. What we have done here? We have done the same thing, you can notice that here (Refer Slide Time: 43:15), I have written A 1 times u bar n plus 1 so that of this since this part was already known to us, so we have use that information that is this.

Now of course, it becomes very massy to keep this as u bar at n plus 1. So, you will find that most people prefer simplify for the sake of analysis and write u bar n plus 1 as is equal to close to u n plus 1 which is a kind of incorrect thing to do and we will show what it actually does but still suppose, we do that then, what we are seeing here that we are getting N now as what? A 2 plus this part.

Now with that approximation that we have made and the 22. We are talking A 2 plus this part A 2 plus A 1 at the current level of iteration, if we assume u bar equal to u at the current.

(Refer Slide Time: 45:14)



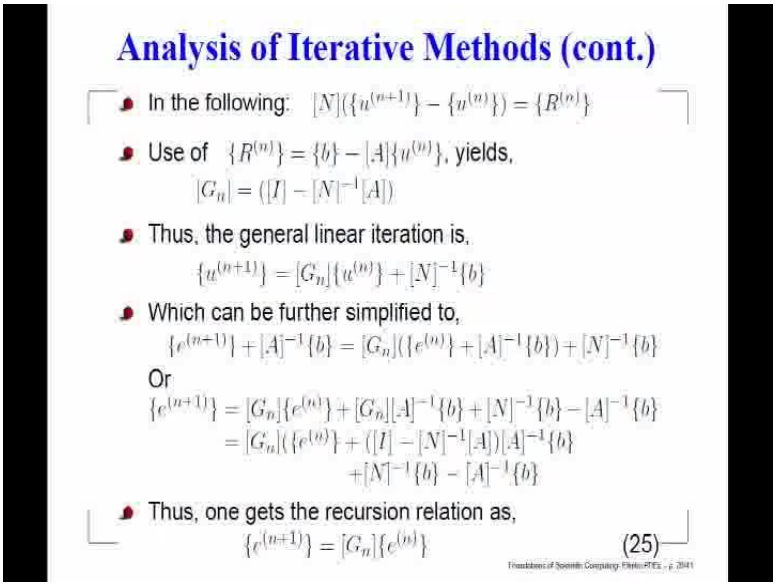
So, this seems to suggest that line Gauss-Seidel would be even more closer to the correct direct solution because, earlier N was A 2 now we have added A 1 to it. So, legitimate question that arises is, how good is this approximation leading to this? We have made this approximation u bar n plus 1 is equal to u of m plus 1 and this is where I decided to invest in time and think about what we are done.

Let us recap what we have done so far. We try to solve this equation. This is our solution strategy; we have replaced A by N and then, driven it by the residue at that level and then operationally, we have gone through this. Then what one could do is, one could write that this u at n plus 1 equal to some G matrix of operating at u n and plus some r n; r n is nothing but this N inverse b in equation 14, so this is your r n. Why did we put all this subscript and superscript? Superscript you already know.

Now, this G n is something like what you call as the gain matrix - we have just now seen while talking about the parabolic equation - we did bring in the concept of the gain matrix. So, whatever we had at the n th iterate; we multiply by the gain to arrive at the new level. So that is where you can see the connection between the elliptic solution methodologies with the parabolic solution that we have just now done, so that is what we do.

However, we must notice that this G matrix may actually depend on the level at which we are working that is why we have purposely added a subscript n.

(Refer Slide Time: 47:40)



Now, the same way the residue that we are calculating here this N inverse b also could become iteration dependent that is why to cover up the possibility, we have called it as power of n. Now, this is the way we have define the error; e of n as what we have at the current level u of n minus the actual solution; actual solution is A inverse b. So the current iterate minus the exact solution that is what we are calling as the error at that level.

What we could do is, we could insert all this information, so now you can see very clearly G n is nothing but, I minus N inverse A and this is our general linear iteration. Why we are calling linear? Because the new iterate depends linearly on the old iterate, so that is as simple as that.

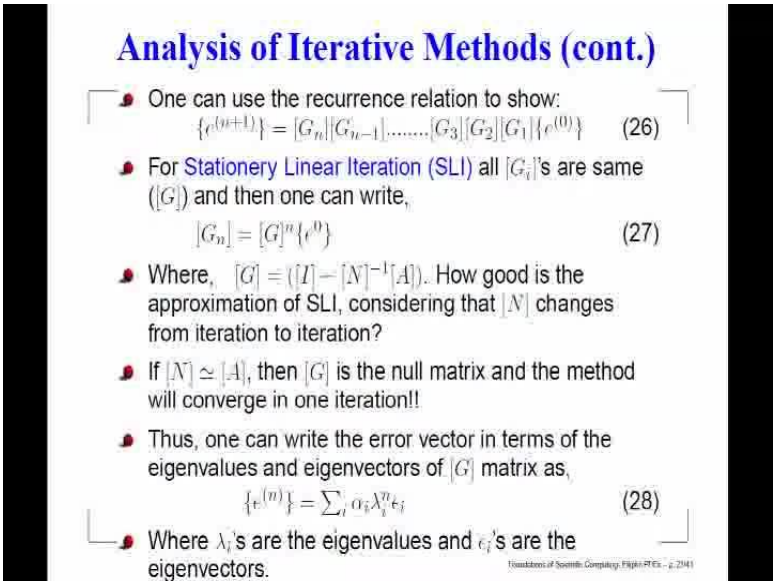
Now what we could do is, I could subtract from this exact solution or I could write the current iterate as the error plus the exact solution. So u n plus 1, I have written it as e at n plus

1 plus A inverse b. Same way on the right hand side, u n has been replaced by e n plus A inverse b. So, do a little bit of manipulation and what you find that e n plus 1 is G n e n plus this part; from this part we are going to get G n A inverse b and we have this N inverse b at the end minus this A inverse b has been brought in here.

So, we could do this manipulation here because, in this I have kept it G n as it is, but here instead of G n I have written I minus N inverse A. So, G into A inverse b is equal to I minus N inverse A; the whole thing is multiplied by A inverse b and then, you can very clearly see this part N inverse A A inverse b will be N inverse b with a negative sign that will cancel with this and this first one that is A inverse b will cancel with this. So, we get a very clean recursion relation here (Refer Slide Time: 49:38).

Now that also would convince you that is why we called it as a gain matrix because, if I have the error at the n th level, I just simply multiply by the gain matrix to get the error at the next level, so that is what we do.

(Refer Slide Time: 50:20)



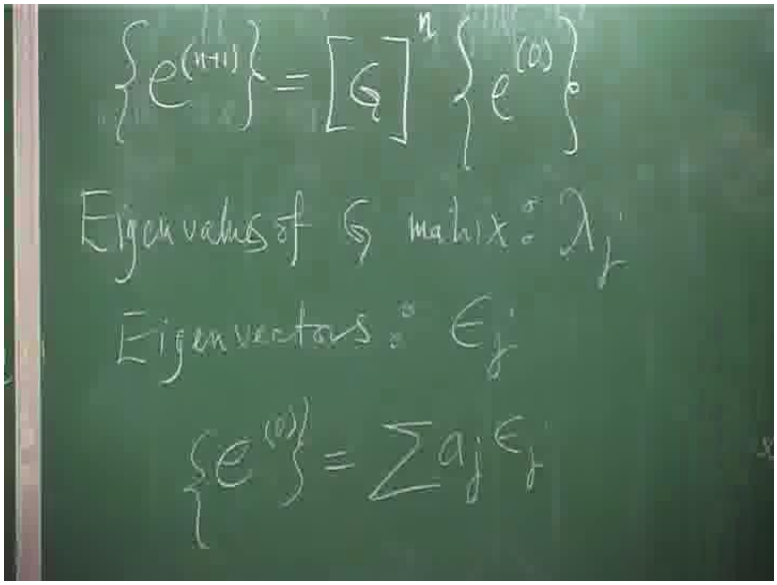
Now, if we keep looking at it since, e n plus 1 was G n into e n. So, e n I could write it as G n minus 1 into e n minus 1 and I could cascade down through that list and then, I could get the error at the n plus 1 th level in terms of the error that we have initially incurred that is what we have called here as e 0.

Now, in the analysis of iterative method what is usually done is an assumption is made that all this GIs are same and such an assumption leads to what we call as a Stationary Linear Iteration SLI. If we have a stationary linear iteration then that G n matrix, well I think I have done a pretty bad job here this is this e 0 should not be here all that you are saying here is G of n is nothing but the G raise to the power n. So that e n plus 1 is G to the power n into the initial error; I have lots of corrections to make.

Now we understand, if we do not we will see it shortly that this N matrix that constitute G; N inverse A, so this N itself changes from iteration to iteration. So, if N is almost equal to A then of course, G is a null matrix and the method will converge in one iteration which does not happen because, we are looking at iterative method.

(Refer Slide Time: 52:35)

e



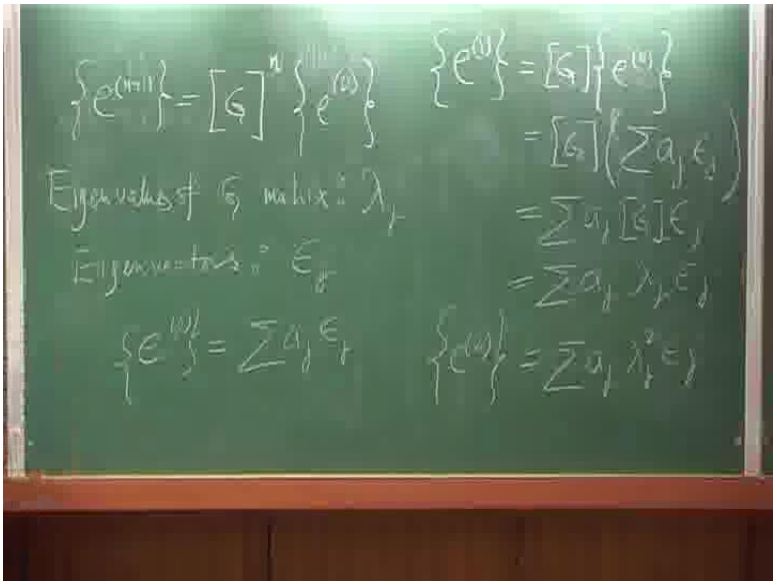
Then what happens is, let me write it down here cleanly. We have written it down there that e error at n plus 1th level is nothing but this raise to the power n, so this is not superscript this is raise to the power n times the error incurred at the first step.

Now of course, what we are doing, we are repeatedly multiplying the initial error by the G matrix. What I could do is, I could actually write this operation in terms of the Eigen value of the G matrix. So, if this Eigen values of this G matrix that we are looking at this, we are writing it as lambda j; a series of lambda j. For the sake of simplicity, let us assume all these Eigen values are distinct; if they are not then, we have to get in something called Chardon canonical form, so which will not like to get into but, let us understand that this is so. So, this

kind of matrix operation would be equivalent to them writing down the initial in terms of this Eigen values because, if I take the size of this matrix quite large then, I will get a spectrum of Eigen value. I have then.

What I could do is, I could write any arbitrary quantity in terms of this Eigen values and Eigen vectors. So, if the Eigen vectors are written as like epsilon j then, the initial error that I could write it as summation of whatever I have written there a j and into epsilon j; I could write that, I could express any arbitrary function in terms of the Eigenvectors set of Eigenvectors.

(Refer Slide Time: 55:17)

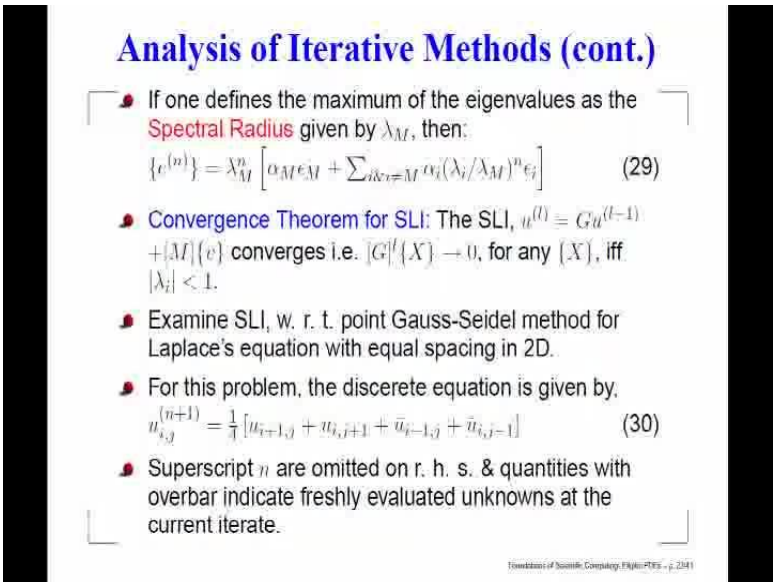


Now, if I am multiplying it by G then, what will happen? If I write e of 1, that will be G times e of 0. Now, G operating on - let me write it like this - a j epsilon j of course, this I can write it as (Refer Slide Time: 55:44). So, this is nothing but your definition of Eigen values and Eigen vectors. So if I get once, I get u 1 in terms of a j lambda j and epsilon j. So if I do e 2, I will be doing this again that means, the lambda G will operate on this again and that will give you again lambda j. So what will happen is, in the same way I could write it like this then, I will get a j lambda j square and so on so forth.

You can take this story as logical conclusion and you will get the error at the n th level would be nothing but, say a j times lambda j to the power n and times the Eigenvector.

So, this is what we see whether the method will converge or not, will be determined by what then? Will be determined by lambda j - the Eigen values. What we are hoping? That as iteration progresses, this error should decay to 0 that can happen when? That can happen when the Eigen values have some specific properties.

(Refer Slide Time: 57:25)



Let us now define one of the defining properties of Eigen values called the spectral radius. Spectral radius is nothing but the maximum Eigen value of that whole set that you have. If that spectral radius is called lambda n then, what we have written there, I could write everything in terms of lambda n, so this is what I get.

From this equation, what you can see that we are writing this error in terms of this maximum Eigen value raise to the power n times the corresponding Eigen vector plus rest of the term, which depends on the ratio of lambda i by lambda m. So, lambda i by lambda n of course, less than 1; by definition lambda is the maximum. You can see when that n goes to a very large value; this second part does not contribute very much; as you along the second part is not playing a greater role. What is determined essentially is by lambda m.

I will start from here and we will talk about this convergence theorem and will get some view of what is happening.