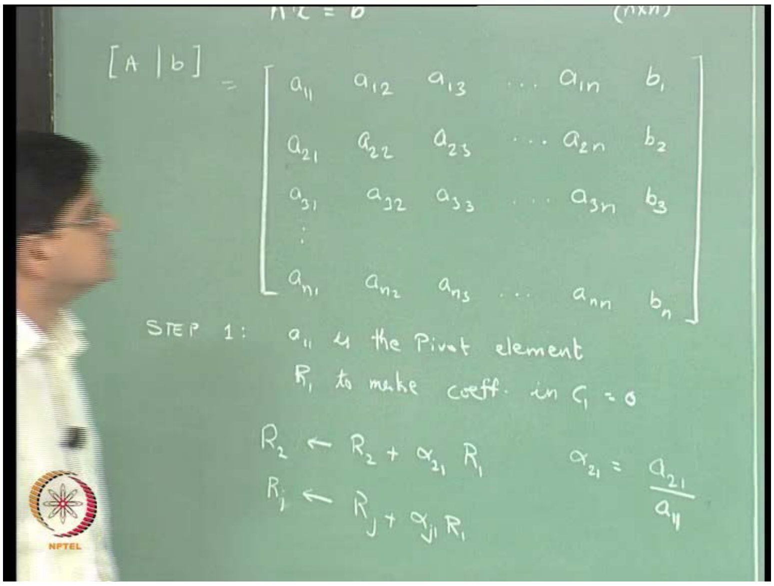
Computational Techniques

Prof. Dr. Niket Kaisare
Department of Chemical Engineering
Indian Institute of Technology, Madras

Module No. # 03
Lecture No. # 05
Linear Equations

Hello, and welcome to lecture 5 of module 3. What I am going to do now? So far, we have been talking of Gauss elimination method in terms of examples. What I am going to do? Now, really is to take a look at an n equation in n unknown problem and then try to see, what is the amount of effort that is required in terms of the number of steps or the number of flops or number of floating point operations that will be required. In order to do to solve the problem using the Gauss elimination method.

(Refer Slide Time: 00:47)



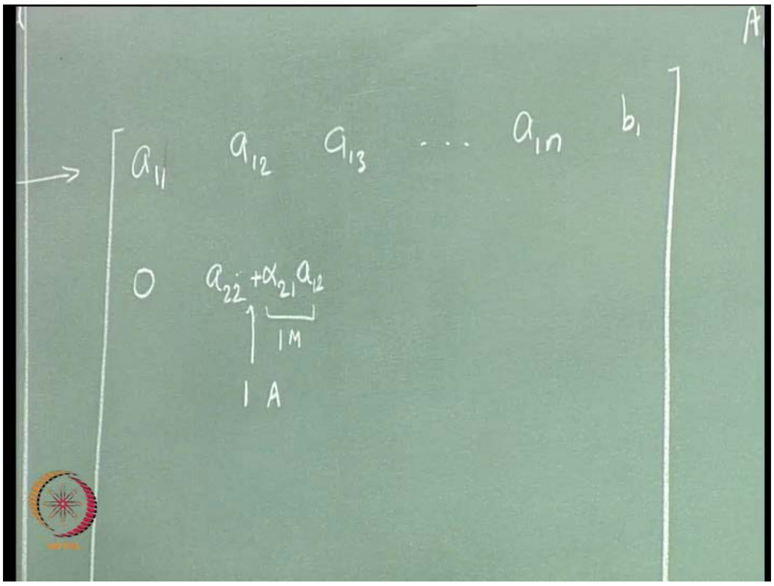
So, again the problem that we have is going to be A x equal to b. And then the first step that we will do is, you will create A matrix by concatenating the columns of the vector b with the matrix a. We will consider on n by n size matrix a, this is a n dimensional problem. And we will although we have discussed about, partial pivoting earlier we are not going to consider partial pivoting at all in this particular discussion simply. Because, partial pivoting the way computer is going to do partial pivoting, it is not going to do any kind of a computation, but it is only going to reassign the locations of the particular rows in its storage places. So, it is very efficient. When we want to do a exchange of rows; using for doing the partial pivoting, it does not take any computational, any significant computational effort to do that.

However, it takes fair amount of effort to do mathematical operations of addition, subtraction, multiplication and division. The older computers were would do addition, subtraction much faster than they could do multiplication and division. Either as a result historically; people have been calculating the number of multiplication, division steps and number of addition, subtraction steps required independently. However, the new computers they are as efficient in doing multiplication, division steps as they are in doing the addition, subtraction step. So, for newer computers the amount of effort required for addition, subtraction is approximately similar; to that required in multiplication, division 1 single instance of multiplication and division.

So, let us look at the A b matrix. And the A b matrix is going to be a 1 1 a 1 2 a 1 3 and so on up to a 1 n b 1 a 2 1 a 2 2 a 2 3 a 2 n b 2. a 3 1 a 3 2 a 3 3 a 3 n b 3 and so on up to a n 1 a n 2 a n 3 and so on up to a n n b n .

In step 1; a 1 1 is the pivot element and R 1 is the pivot row. We will use R 1 to make coefficients in the pivot column equal to 0. So, if we will do row operations in order to make a 2 1 a 3 1 and a n 1 equal to 0. Right and what how we do? That is R 2 if we will replace it with R 2 plus alpha 2 1 R 1. R 3 with alpha R 3 plus alpha 3 1 R 1 and so on. In general, R j will be replaced with R j plus alpha j 1 R 1. Where, alpha 2 1 is nothing but, a 2 1 divided by a 1 1. So, now, let us look at doing the first operation.

(Refer Slide Time: 05:01)

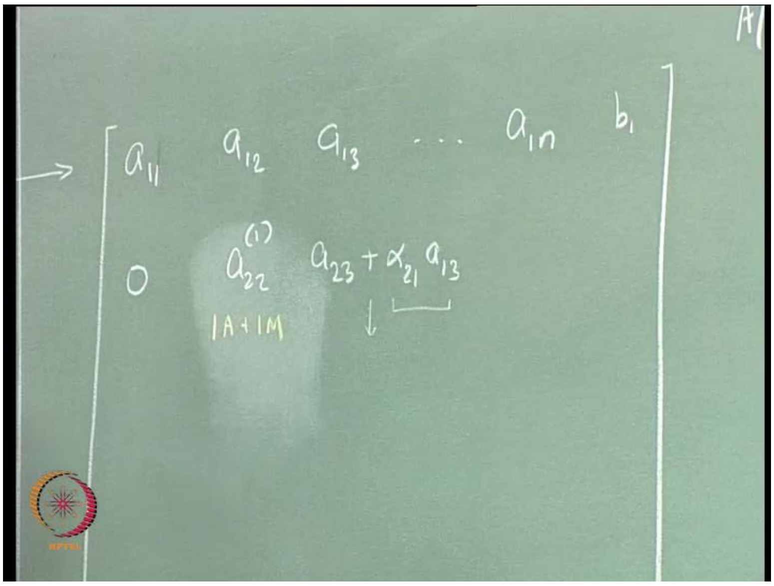


So, when we are doing the first operation the row 1; remains the same there is no change in row 1. So, I will write it as a 1 1 a 1 2 a 1 3 and so on up to a 1 n b 1. The first computation is in computing a 2 1. So, I will write the addition, subtraction steps over here multiplication, division steps over here.

There is 1 multiplication division step; that multiplication division step is to compute this particular coefficient alpha 2 1. So, now, once we have completed alpha 2 1 we will subtract R 2 minus alpha 2 1 into R 1. As, we have said earlier we will not going to do this 1 extra calculation. Because, we know a priory that this number is going to be equal to 0. So, this calculation we are not going to do.

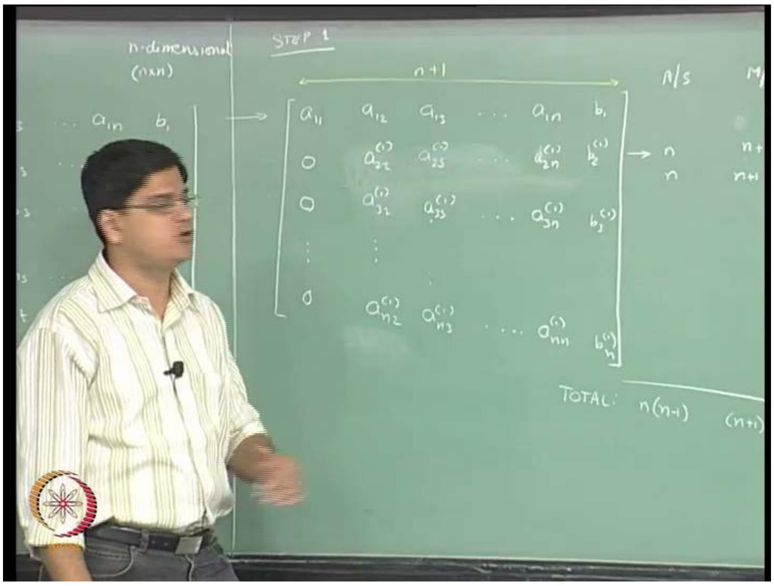
Here, what? This particular element is going to be a 2 2 minus or a 2 2 plus alpha 2 1 multiplied by a 1 2. So, this is 1 multiplication step, and this is 1 addition step. So, there is 1 multiplication and 1 addition step in getting from a 2 2 to a 2 2 sub superscript 1.

(Refer Slide Time: 06:38)



So, we will get this as a 2 2 superscript 1 when we have 1 addition and 1 multiplication step. Likewise, a 1 3 a 2 3 1 is going to be nothing but, a 2 3 plus alpha 2 1 a 1 3. So, this is 1 multiplication step and 1 addition step.

(Refer Slide Time: 07:10)



So, a 2 3 1 is also obtained from; 1 addition plus 1 multiplication step and we repeat this for remaining n minus 1 elements of the matrix a and 1 element of the vector b. Keep in mind, that this distance this number is n plus 1. There are n columns because of, matrix a and n plus 1th column because of the vector b. This computation we are not doing explicitly we are only doing all these computations.

So, in this particular row operation we have the number of addition, subtraction steps are going to be equal to n. The numbers of multiplication, division steps are going to be equal to n. So, for row operation to get row number 2 the total numbers of multiplication division steps are going to be n plus 1. The plus 1 is for computation of that coefficient.

So, this is the row operation in order to get row number 2. Now, let us see. What happens in computing row number 3? in row number 3 again, we get 0 over here; and we will compute a 3 2 1 and again a 3 2 1 is going to be equal to a 3 2 plus alpha 2 1 multiplied by a 1 2 .So, this is what? A 3 1 multiplied by alpha 1 2. So, we need 1 computation again in computing alpha 3 1 which is nothing but a 3 1 divided by a 1 1.

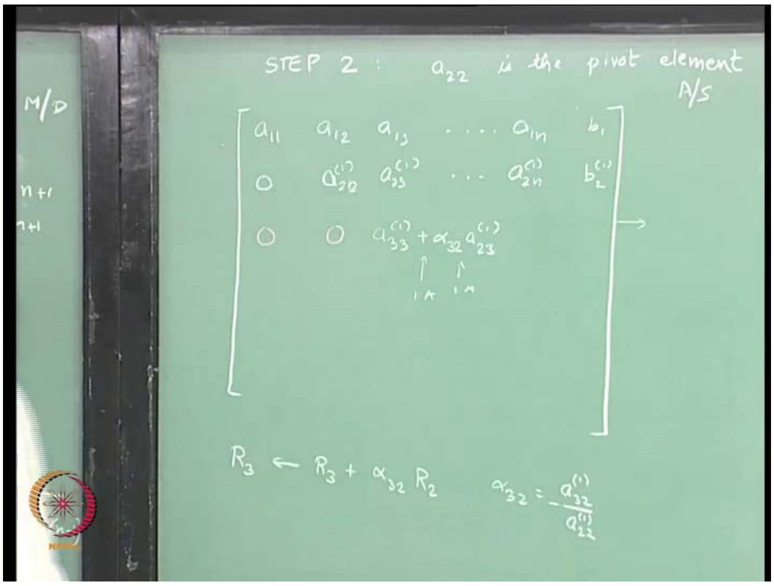
So, that is 1 multiplication and division steps and then to get a 3 2 1 we will have 1 multiplication step and 1 addition step. Here, we will have 1 multiplication 1 addition step so and so forth exactly in the same manner as we added for row number 2.

So, the number of steps required over here, are going to be n addition subtraction steps and n addition multiplication division step plus 1 more multiplication and division step to get alpha 3 1. And as you see, what pattern we are following? Is that for each and every row the number of steps that are required are exactly the same? As a result of this, we what will get is a n 2 1and so on up to a n 3 1 a n n 1 b n 1 b 2 1 a 2 n 1 a 3 3 1 and so on up to a 3 n 1 and b 3 1.

All these are obtained; through n addition subtraction step n multiply and n plus 1 multiplication division steps. And let us see, how many rows are there to do this row operation there is 1 2 3 4 and so on up to n minus 1 rows. The total number of rows are n the first row is being unchanged only the n minus 1 rows are being changed. So, what we are going to half is n plus n plus n n minus 1 times and n plus 1 n plus 1 n plus 1 n minus 1 times.

So, the total number of addition subtraction steps is going to be n multiplied by n minus 1 and the total number of multiplication division steps are going to be n plus 1 multiplied by n minus 1. This is at the end of step 1.

(Refer Slide Time: 11:20)



Now, we will go ahead and see, what happens in step 2? In step 2 a 2 2 is the pivot element. Keep in mind that, when we write algorithm for Gauss elimination method. We do not create separate matrices separate matrices at each step that would be extremely inefficient to do. What we are going to do? Is as I have written over here, we as soon as this particular computation is done. The original number that we had over here a 2 2 is going to be replaced with replaced by a 2 2 1.

So, although I am showing the superscript 1 over here and so on. In the computer memory there is only 1 a b matrix. So, that resides in the computer's memory and that a b matrix keeps changing when every time you do any row operation.

As a result, as soon as this particular operation is done. The old value has already been replaced by a 2 2 superscript 1. The superscript I am using just so, that an able to distinguish the original values from the newly computed values for no other reasons. I am using these superscript keep in mind that you will use the same a matrix in order to do all the row operations and all the column operations the results will be stored in the same a matrix.

So, now let us see what happens at the end in step 2 not at the end. But in step 2, still we will have this matrix again and then we have addition subtraction steps multiplication division steps. Row 1 unchanged row 2 unchanged. So, I will just copy those rows as they are a 1 1 a 1 2 a 1 3 and so on a 1 n b 1. 0 a 2 2 1 a 2 3 1 and so on up to a 2 n 1 b 2 1.

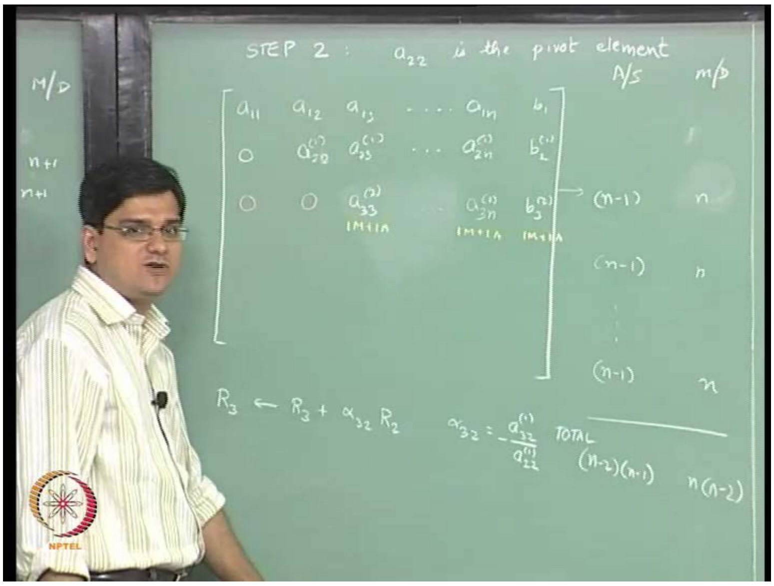
The next step, we are going to do is R 3 is going to be replaced by R 3 plus alpha 3 2 multiplied by R 2. Now, this step first requires computation of alpha 3 2, alpha 3 2 is nothing but a 3 2 divided by a 2 2 negative of that will again; put, superscript 1 again this putting the superscript 1 is really optional because, from a computational point of view is the same variable that is storing that that particular number.

So, we have 1 multiplication division step come in over here, 1 multiplication division step come in row 3. Now, because of this particular value being 0 no matter what row operation we are going to do this value you have a 3 1 value is not going to change. So, this 3 1 value is 0. That we had over here, will remain 0 it is not going to change we are not going to waste 1 operation in computing this particular value. Same thing, as we had done before this a 3 2 value after the first row operation because of the choice of alpha 3 2 is going to definitely become equal to 0.

So, those values are not going to change. So, the unchanged again I am going to over here. I am going to just put them with the red color chalk. So, these 2 values are not computed we only we just keep this value unchanged and we just replace the original value over here with a value 0. This 3 3 is going to be equal to a 3 3 1 plus alpha 3 2 multiplied by a 2 3 1.

So, this again in the same manner as we have been doing before this is; how the computation goes and this will have 1 multiplication step 1 addition step. And as a result of 1 multiplication and 1 addition step we will get this value as a 3 3 2.

(Refer Slide Time: 16:16)



So, a 3 3 2 has a resulted from 1 multiplication plus 1 addition step. And so on, we will go on up to a 3 n 2 and b 3 2 again for a computing a 3 n 2. We will have 1 multiplication plus 1 addition step and 1 multiplication plus 1 addition step over here as well.

Now, there were total number of columns are n plus 1 and 2 of the columns we have not done any addition, subtraction, multiplication, divisions steps at all none of the all algebraic manipulations are done over here.

We have kept this number as it is and we have replaced this original number with 0; which does not require any significant computational effort. As a result of this, the total amount of effort is going to be for n minus 2 multiplications and n minus 2 additions there are n plus 1 columns n plus 1 minus 2 is actually going to be n minus 1 not n minus 2.

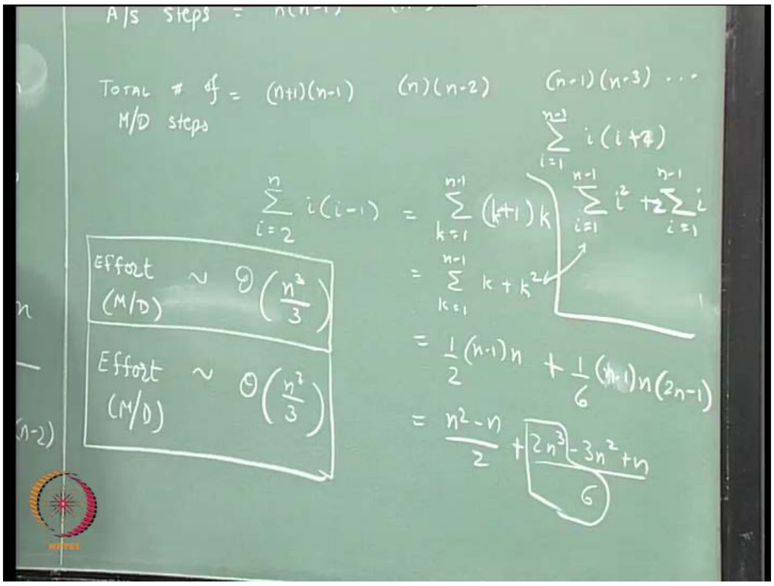
So, we will have n minus 1 addition subtractions n minus 1 multiplication divisions and plus 1 more multiplication division to get alpha 3 2. So, we will have the total number of operations is going to be equal to n. Then, we will do this for row number 4 for row number 4 again we will have exactly as we had it had in this particular case in this particular operation step 2 operations. In the same way, we will in the step 3 as well as these 2 numbers are going to be 0. And there will be 1 multiplication 1 addition steps in the remaining n minus 1 computation.

So, as a result again we will have n minus 1 number of addition subtraction steps and n multiplication division steps and this particular row and this will go on up to here. So, the number of times we have this is the total number of rows are n and the rows that are changing are row number 3 to row number n. So that means, n minus 2 rows these operations we are conducting.

So, the total number of operations that we get is n minus 2 multiplied by n minus 1 and in this particular case the total number of operations are going to be n multiplied by n minus 2. So, in this case the total number of operations where n multiplied by n minus 1 and n plus 1 multiplied by n minus 1. In this particular example, we have n minus 2 multiplied by n minus 1 and n multiplied by n minus 2 operations.

And we will continue this, so on and so forth for n minus 1 steps total and we will so in the next step, we will have n minus 3 multiplied by n minus 2 operations of addition subtraction and n minus 1 multiplied by n minus 2 of multiplication divisions.

(Refer Slide Time: 19:42)



So, total number of addition subtraction steps are going to be equal to we go back over here, n multiplied by n minus 1 then n minus 2 multiplied by n minus 1 n minus 3 multiplied by n minus 2 and so on and so forth. So, n n minus 1 n minus 1 n minus 2 n minus 2 n minus 3 and so on this is in step 1. Step 2 and so on up to step n minus 1 and in step n 1 n minus 1 we will have 2 multiplied by 1 number of steps.

Total number of multiplication and division steps. So, total numbers of multiplication and division steps are going to be n plus 1 multiplied by n minus 1 n multiplied by n minus 2 and so on and so forth. So, this as a result what we are going to get is the total number of addition subtraction step is just some of these; and the total number of multiplication division steps is just some of this particular thing.

So, in other words the total numbers of addition subtraction steps are going to be summation i multiplied by i minus 1. Where, I go from 2 to n. Because, we have n minus the total number of steps are n minus 1 as a result this goes from 2 to n 2 to n is going to be the total number is going to be n minus 1. And this we can write this as summation k equal to 1 to n minus 1. So, where k is going to be just i minus 1.

So, this we can write this as i minus 1 multiplied by i minus 2 .Where, i multiply i plus 1 multiplied by I, k plus 1 multiplied by k. So, from k going from 1 to n plus 1 and this is going to be equal to summation of k plus k squared. Summation k equal to 1 2 n to n minus 1 of k is going to, it is nothing but 1 plus 2 plus 3 up to n minus 1; if we have up to n the result is going to be n multiplied by n plus 1 divided by 2; if we have up to n minus 1 is going to half of n minus 1 multiplied by n.

And summation of k square is from k equal to 1 to n minus 1 is going to be equal to 1 by 6 multiplied by n minus 1 multiplied by n; multiplied by 2 times n minus 1 minus plus 1 2 times n minus 1 plus 1 and that is going to be 2 n minus 2 plus 1 2 n minus 1. Which, we can write this as n square minus n by 2 plus 2 n cubed. So, n minus 1 2 n minus 1 is going to be 2 n squared minus 3 n plus 1 multiplied by n is going to be 2 n cube minus 3 n squared plus n divided by 6.

So, these are going to be the number of addition subtraction steps. If these are the number of addition subtraction steps that we have the leading term over here. So, let us say, if n is very large say n is moderately large say n is ten; in even if n is ten n cube is 1 whole order of magnitude larger than n squared and 2 orders of magnitude larger than n.

So, as the number of elements start increasing or the number of variables start increasing, n cube starts becoming larger much faster than n squared or n as a result the total number total amount of effort required is actually governed by this particular term.

So, the number of addition subtraction steps are effort in terms of, the total number of addition subtraction steps is of the order of n cube divided by 3 in case of Gauss elimination method. And the total number of multiplication division step is n minus 1 n plus 1 multiplied by n minus 1 n multiplied by n minus 2 and so on and so forth. Which, we will be able to write it as summation i equal to 1 to n minus 1 and this is i multiplied by i minus 2.

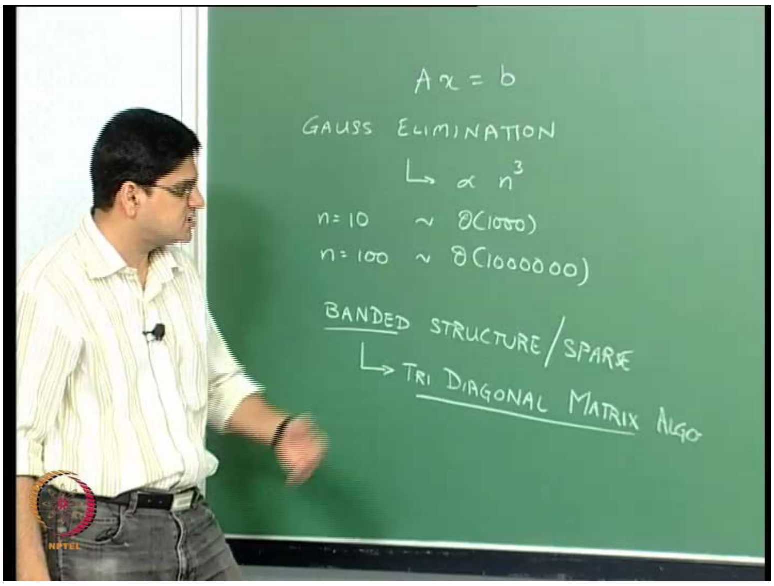
So, i multiplied by i plus 2. So, at 1 it is going to be 1 multiplied by 2. So, 1 multiplied by 2 plus 1 2 multiplied by 4 plus 3 multiplied by 5. So, on and so forth up to n minus 1 multiplied by n plus 1. So, this we can expand this as summation i equal to 1 to n minus 1 i squared plus 2 times summation i equal to 1 to n minus 1 2 i. And this summation of I squared is the same as summation of this i squared.

So, again the order of effort is going to be determined by summation of i squared. Where, i go from 1 to n minus 1. So, for multiplication division steps also effort is of the order of n cube by 3. So, in summary what we have for Gauss elimination method is that the amount of effort required in solving a problem an x equal to b scales as the cube; not square, not linearly but as the cube of the number of variables n.

So, when we start with an equation a x equal to b. Where, x is an n dimensional vectors and we have n equations, if we were to use the Gauss elimination method the amount of effort is going to be of the order of 2 n cubed divided by 3. Because, we have n cube divided by 3 as the multiplication division steps and n cube divided by 3 as the addition subtraction steps.

So, that is the effort involved in Gauss elimination in exactly, the same manner we can also compute the effort involved in back substitution; and if we were to do that we will find out that the back substitution step scales as n squared and not as n cube.

(Refer Slide Time: 27:59)



Whenever, we have any problem of the sort A x equal to b we can employ the method such as Gauss elimination to solve a general problem. what we just saw a few minutes back, is that Gauss elimination is highly computationally intensive method it is computational intensive because, the number of computation required for Gauss elimination are proportional to n to the power 3.

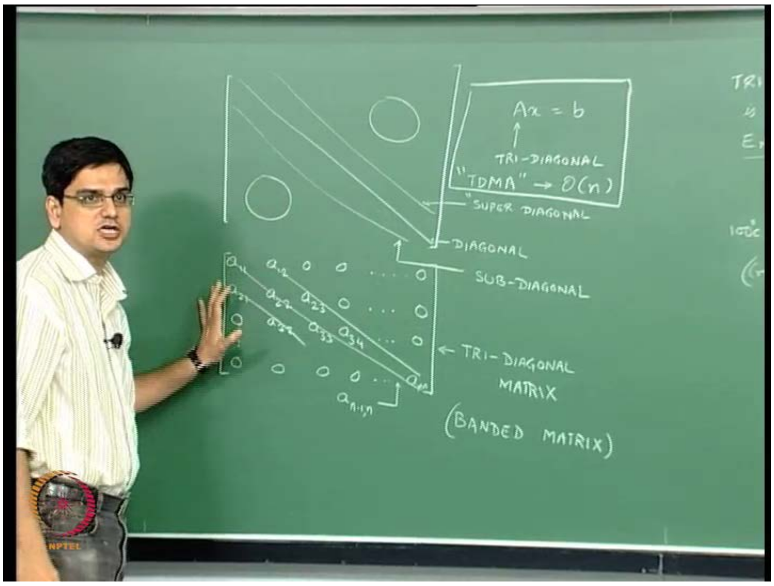
Where, n is the number of variables in the vector x that we are solving for. So, for example, if n equal to ten we have the number of computations that are required in order to use Gauss elimination are of the order of 1000. However, from n equal to ten we go to a slightly larger problem with say n equal to say 100. This particular the number of computations that are required increases from thousand to 1 million and that is because, Gauss elimination scales as n to the power 3.

Now, if for from n equal to 100 if we go to n equal to 1000 the amount of computation required is going to be ten to the power 9. So, as you as we increase the number of the number of equations that we are solving the amount of effort required in Gauss elimination increases very rapidly. Now, for a general purpose problem A x equal to b we do not have a choice because, Gauss elimination turns out to be 1 of the more efficient ways of solving a general purpose problem A x equal to b. However, a lot of problems of interest to engineers' chemical engineers, mechanical engineers and so on are not a general purpose problem they have a specific structure.

And they have a structure what is known as a banded structure or sparse structure. So, this matrix a is not a full matrix, but it has a certain type of structure and where does that structure comes from I will come to that in a couple of minutes, but because it has a certain structure; there are certain properties of that structure that we can actually explain. And that is exploited in some of the more efficient ways of solving those are banded solvers as they are called.

A specific example that we are going to take of the banded solver today is called a tri diagonal matrix algorithm. And the tri diagonal matrix algorithm is applicable to, what are known as tri diagonal matrices? So, let me come to what tri diagonal matrices really or and how this tri diagonal matrices are come into existence for a typical problem that we see in chemical engineering.

(Refer Slide Time: 31:16)



So, the tri diagonal matrix structure is of the form. So, this is what we have the matrix a is going to be, in the tri diagonal structure the diagonal elements of the matrix may be nonzero, 1 more set of elements just below the diagonal may be nonzero.

One more set of elements just above the diagonal may be nonzero whereas, rest all the elements in this matrix are going to be 0. An example of this could be for example, you can have a matrix say a 1 1 a 1 2 0 0 and so on up to 0 a 2 1 a 2 2 a 2 3 0 and rest everything else is going to be 0.

Now, this particular guy is 2 locations of the diagonal therefore, this guy is going to be 0. Then we will have a 3 2 a 3 3 a 3 4 and rest everything will be 0 and so on we will have 0 0 0 0 and so on up to a n n. And there will be over here, this term is going to be a n minus 1 comma n rest everything in this particular row is going to be 0.

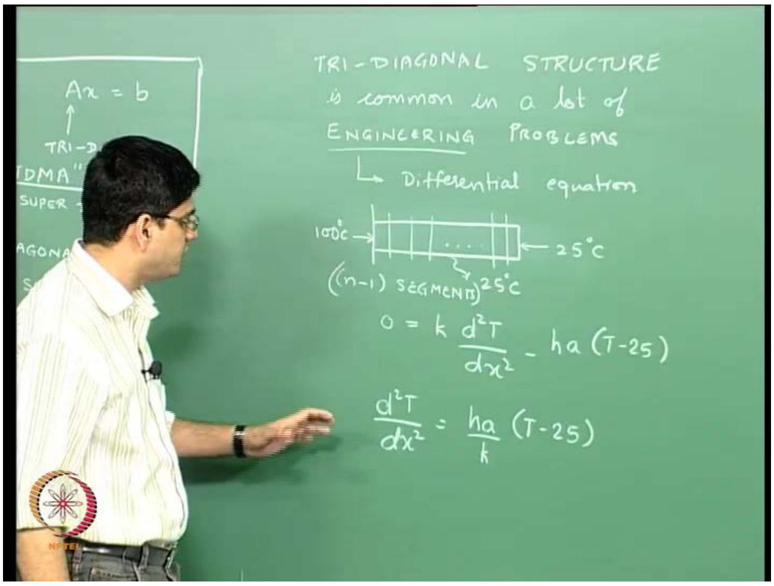
So, this is a tri diagonal structure as we can see over here, this particular diagonal may be nonzero this super diagonal element may be nonzero and this sub diagonal elements may be nonzero everything else in this structure is 0s.

So, again this is called a diagonal; this is going is, what is known as super diagonal? And this is going to be sub-diagonal. And if, we have 1 diagonal element 1 super and 1 sub-diagonal element; what we actually get is a tri diagonal matrix. And going for the taking this concept of tri diagonal further, we could we could basically also have a banded diagonal matrix or a banded matrix. Where there might be there would be 1 diagonal element and more than 1 super diagonal elements and more than 1 sub-diagonal elements.

So, for example, there might be 2 super diagonal elements 2 sub-diagonal elements and rest everything else might be 0. Now, these are special structures because we know a priory that everything over here is going to be 0 and everything over here is going to be 0.

The computational load for this can be reduced quite significantly compare to a general Gauss elimination method. So, if we were to solve problem A x equal to b where a is a tri diagonal structure. Then, we can use the T D M A matrix T D M A method tri diagonal matrix algorithm. As, I said earlier you can use this T D M A method to reduce the amount of effort from n to the power 3 to n to the power 1. And I will show that in a few minutes.

(Refer Slide Time: 35:11)



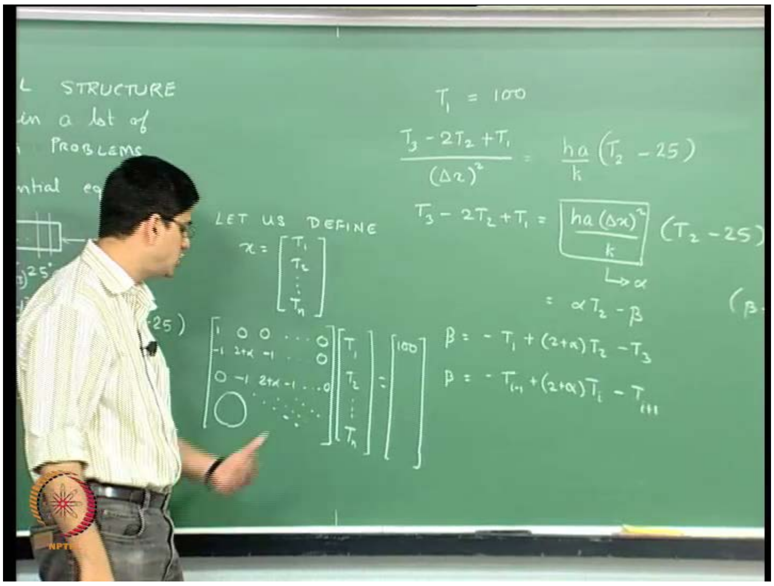
It is common in lot of engineering problems; especially, when we are going to express a model for an engineering problem in terms of a differential equation. We will come to this in more details in module 7 module 8 and module 9 of this particular lecture series.

Let us consider a rod that is losing heat to the surroundings and let the surroundings be at 25 degree Celsius. 1 end of the rod let say is kept is heated and kept at say 100 degree Celsius and let us, say the other end of the rod is kept at 25 degree Celsius and we want to find out how the temperature varies in this particular rod.

Now, this is a straight forward engineering problem. This, results in what is known as ordinary differential equations boundary value problem and under steady state this equation ends up the 0 equal to k d square T by d x square minus h a multiplied by T minus 25 equal to 0.

So, this is the conduction term and this is the heat loss term that comes over here. Now, what we will we can rewrite this entire problem as d square T by d x square equal to h a by k T minus 25. So, this is going to be the overall differential equation, where 1 end of the rod is kept at 100 Celsius the other end of the rod is kept at 25 degree Celsius. And when we rewrite this particular equation, what we are going to do this? We are going to discretize this particular rod in multiple segments. And we can discretize this in n minus 1 segment and when we do that we will result, what we will get this we will get n equations in n unknowns. And those equations and I am not going to derive them now, but I am just going to state then very quickly we will do the derivation in module 8 of this particular course module 8.

(Refer Slide Time: 38:15)



Lecture 2 onwards we will take that derivation. Our T 1 that is temperature at the first end I will going to be equal to 100, for our second segment we are going to use basically, this equation and we are going to expand it using the numerical using the derivatives definition at this particular location.

So, if this location is location 2 then d square T by d x square is going to be T 3 minus 2 T 2 plus T 1 divided by this let us call this as delta x. So, divided by delta x squared. So, what we will have is we will have T 3 minus 2 T 2 plus T 1 divided by delta x squared is going to be equal to h a divided by k multiplied by T 2 minus 25. And I take, delta x squared on to the other side I will be ah I will basically get T 3 minus 2 T 2 plus T 1 equal to h a delta x squared by k T 2 minus 25.

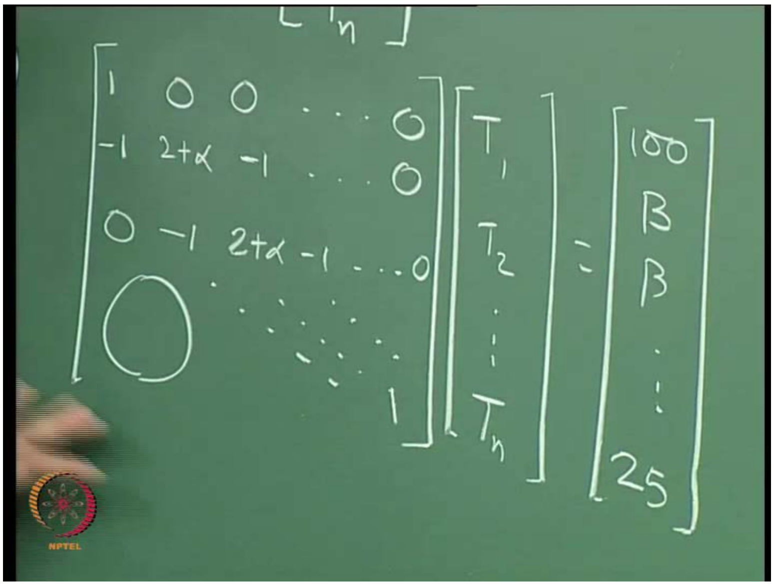
Let us call this guy as alpha. So, we will get this as alpha T 2 minus alpha multiplied by 25 let us call 25 alphas as nothing, but beta. So, where beta equal to 20 alpha. So, this is what we will get we will rearrange this we will take T 2 on to or rather we will take b beta on to the other side and these guys we will take, on to the right hand side and we will get beta is going to beta is going to be equal to minus T 1 plus 2 plus alpha T 2 minus T 3 is going to be equal to beta.

So, this is what we are going to get, when we redo this particular procedure for the third element, we are going to get beta equal to minus T 2 plus 2 plus alpha T 3 minus T 4 equal to this left hand side. So on and so forth or we will get beta equal to minus T i minus 1 plus 2 plus alpha times T i minus T i plus 1.

So, now our overall structure of our matrix is going to be determined. As follows let us define x equal to nothing but T 1 T 2 and so on up to T n, then our equations can be written in the form like this. So, the first equation is T 1 equal to 100 or it is going to be 1 0 0 and so on up to 0 multiplied by T 1 T 2 and so on up to T n equal to 100. Our second equation is going to be minus 1 2 plus alpha minus 1 and rest everything is going to be 0.

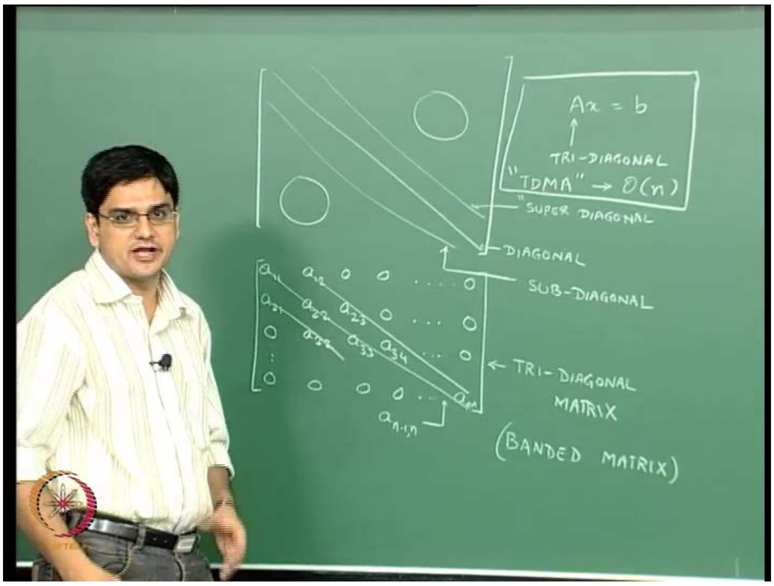
So, it is going to be minus 1 2 plus alpha I will just rewrite this. So, that this enough space for me to write everything 2 plus alpha minus 1 and rest everything is going to be 0 and over here we will have 0 minus 1 2 plus alpha minus 1 and then everything is going to be 0 .And this will follow in the sub-diagonal element this will follow in the diagonal element, this will follow in the super diagonal element every things else will be 0s over here, and the last equation is going to be T n equal to 25.

(Refer Slide Time: 42:52)



Which is going to be just this 1 multiplied by T n equal to 25 and all these guys are going to be populated by the left hand side which is going to be beta. So, this is going to be the overall structure our matrix has the tri diagonal structure a tri diagonal structure.

(Refer Slide Time: 43:12)

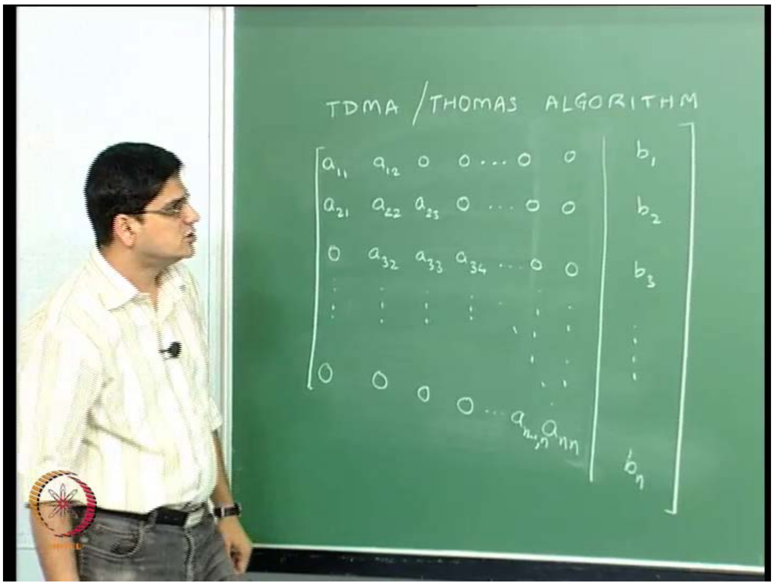


If we come and look over here, this was what the tri diagonal structure. So, we have the diagonal element the super diagonal element and the sub diagonal element all the diagonal elements are 2 plus alpha, except for a 1 1 and a n n all the super diagonal elements are going to be equal to minus 1.

All this sub diagonal elements are again going to be minus 1 except the last guy which is 0. So, this is the tri diagonal matrix structure that comes in because of the heat conduction problem.

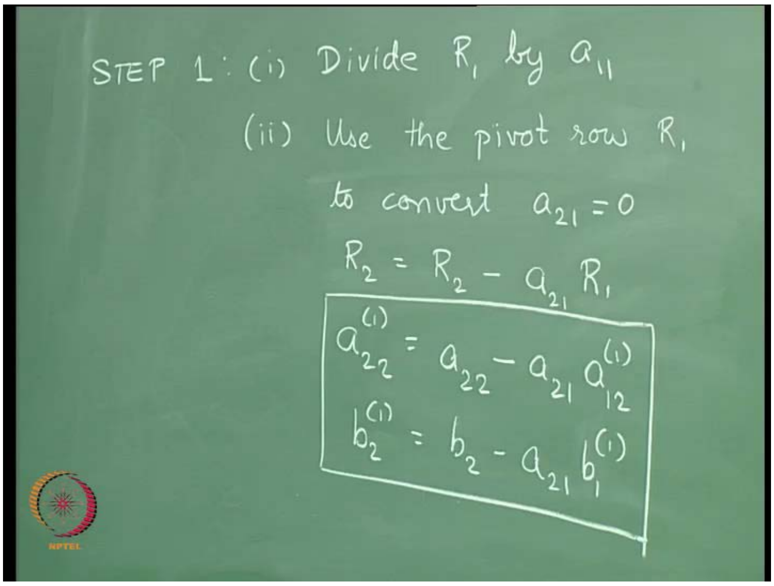
Similar tri diagonal structures you will get from, a plug flow reactor problem or a fixed bed reactor problem or a membrane problem. Where, we are going to if we were to model say reverse osmosis process in a membrane will get a similar type of model and so on and so forth. So, we are going to get tri diagonal matrix type of structures in a lot of problems of interest to chemical engineers. We are going to cover them in module 8 in much more details and how to get to this tri diagonal structure.

(Refer Slide Time: 44:18)



Now, the first step in Thomas algorithm is going to be to make this particular element this pivot element equal to 1 by dividing the entire row by that particular element.

(Refer Slide Time: 44:34)



So, we will write step 1 the first part is divide R 1 by a 1 1 that is what we are going to do the next step in case of Gauss elimination was to use the pivot element or not the pivot element, but the pivot row. So, to use the pivot row R 1 to get 0s in c 1 below the diagonal. So, this was the second step the second part of step 1 in Gauss elimination.

Now, let us go and look at what kind of structure we have in the tri diagonal matrix now in the tri diagonal matrix row 2 has a non 0 element row 3 does not have a non 0 element row 4 does not have a non 0 element up to row n. None of these, rows have non 0 elements in there column 1 below the diagonal only 1 row has non 0 elements that is the first thing to keep in mind.

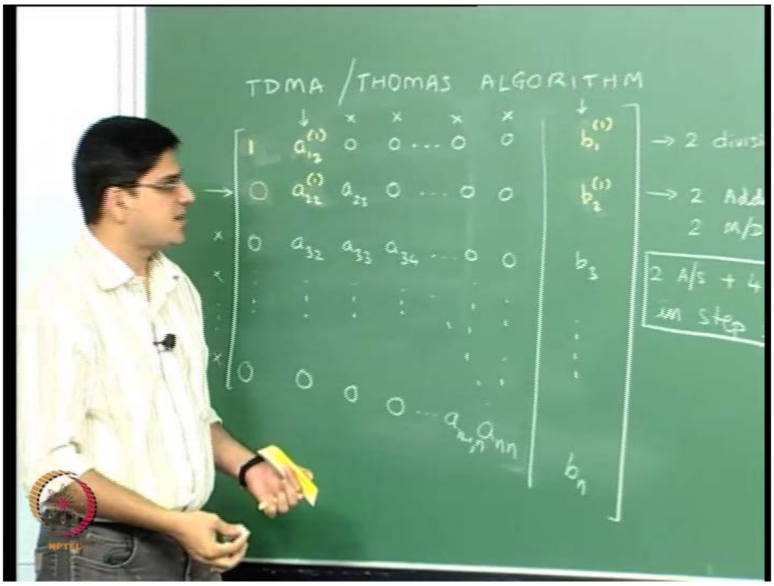
The second thing to keep in mind is that in row 1 this particular element is going to be non 0 this element is 0.

So, this guy minus anything multiplied by this guy is going to remain this minus anything multiplied by this is going to remain this and so on and so forth. So, for wherever I am marking this x marks. We are not going to need to do any calculations at all and we will need to do 1 more calculation over here. So, what it means in order to row use the pivot row we will use the row R 1 to get 0 in this particular element that is the only thing that, and for that is needed and once we get 0 in this particular element. The only element that is going to change is a 2 2 and b 2 none of the other elements are going to change at all. So, what we are going to do over here, is we are going to rewrite this use pivot row R 1 to convert a 2 1 equal to 0. That we are going to do by writing R 2 equal to R 2 minus a 2 1 multiplied by R 1 and unlike Gauss elimination.

There are not going to be n steps; in this there are only going to be 2 steps because, only 2 elements are non 0 a 1 2 and b 1 these are the only non 0 elements. Because of, this what we are going to have is a 2 2 new is going to be equal to a 2 2 old minus a 2 1 multiplied by a 1 2 new this is going to be 1 equation. And the other equation is going to be b 2 is going to be equal to b 2 old minus a 2 1 multiplied by b b 1 old.

So, these are the only 2 equations or only 2 expressions that we are going to actually, solve for when we do when we are going to use R 1 in order to get the pivot element, but the elements in the pivot column equal to 0.

(Refer Slide Time: 48:48)



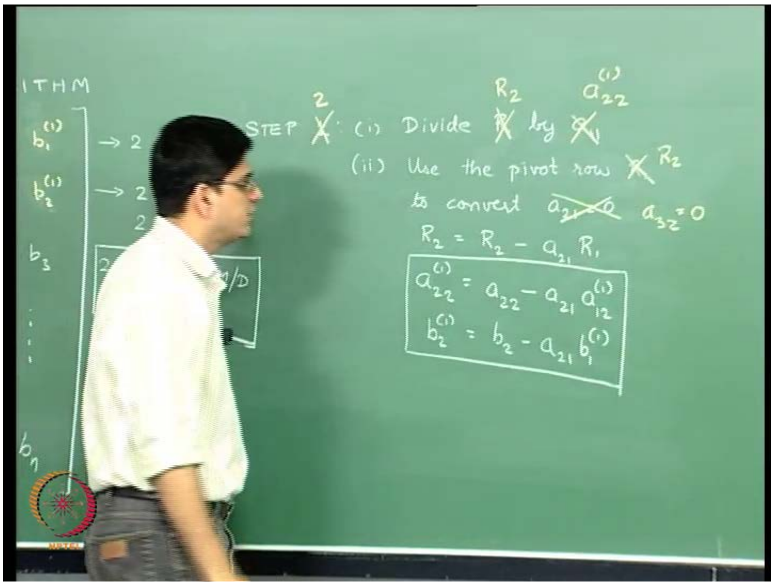
So, what does that amount to do? Is this guy when we divide throughout by a 1 1 becomes equal to 1 a 1 2. Becomes equal to the guy is going to be a 1 2 divided by a 1 1 which we will write a 1 2 1 and b 1 becomes b 1 1.

So, this involves 1 division and the second division. So, step 1 involves 2 divisions the first part involves 2 divisions, what we do next is this particular row minus a 2 1 multiplied by this row by doing that this particular element gets converted into 0. We know, it is going to get converted into 0. So, this computation is not done a 2 2 is going to be equal to as we had seen earlier if we look at this equation a 2 2 is going to be a 2 2 minus this particular product.

So, there is 1 multiplication and 1 addition subtraction step. So, that is going to convert a 2 2 into a 2 2 1 a 2 2 1 is nothing, but a 2 2 minus a 2 1 which was the old value over here multiplied by a 1 2 1. None of these numbers change these numbers remain as they are and b 2 gets change to b 2 1. And again using the equation that I had shown earlier b 2 1 remember is nothing, but b 2 minus a 2 1 multiplied by b 1.

Which involves 1 addition subtraction and 1 multiplication division step this involves 1 addition subtraction plus 1 multiplication division step. So, we have 2 additions or and 2 multiplication division steps. So, step 1 involves 2 addition subtraction steps plus 4 multiplication division steps in step 1. Next, what is going to be step 2? Our step 2 is going to use this particular row as the pivot row. So, in step 2 rows 2 becomes the pivot row.

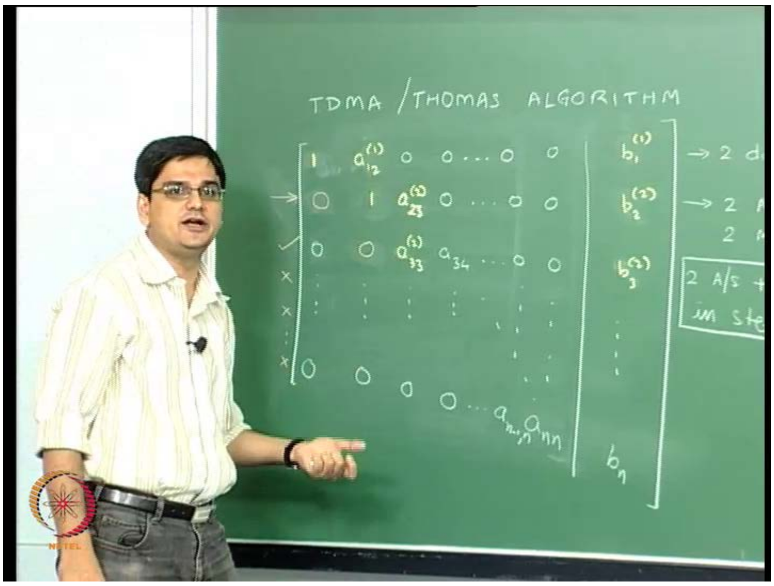
(Refer Slide Time: 51:38)



Now, let us look at, what happens in step 2? So, in step 1 row 1 was the pivot row instead of that in step 2 row 2 is the pivot row and a 2 2 1 is the pivot element.

Now, we are going to use the pivot row instead of R 1. We will use the pivot row R 2 to convert not a 2 1 equal to 0, but a 3 2 equal to 0 and this is going to change appropriately.

(Refer Slide Time: 52:13)



Now, again let us go back and look at what that is going to involve we are going to use this particular row to get 0s in this particular column below the diagonal. So, if this is the diagonal we need to get 0 over here, and everything else below that is 0. So, the only computation that, we are going to do is for row 3 there is no computation that will be done for row 4 row 5 up to row n likewise this a 3 2 itself on its own is going to become 0.

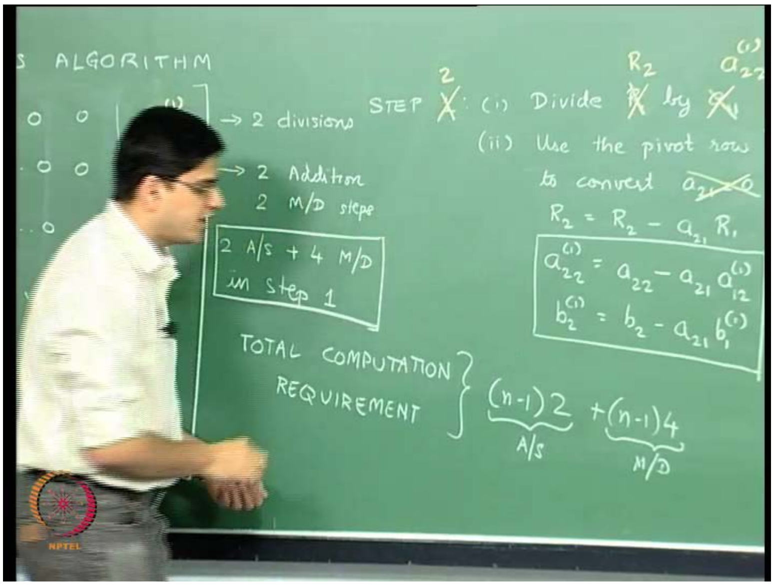
Because, we are going to multiply divide we are going to subtract from R 3 this particular row appropriately. So, this on its own is going to become 0 a 3 3 will change. Because, this is a non 0 element and a 3 3 2 is going to be nothing but a 3 3 minus a 3 2 multiplied by a 2 3. Exactly, the step that we did over here, a similar step will be done over here a 3 4 does not change because the guy over here is 0 rest everything is 0.

So, none of these values change and the only other value that changes is b 3 becomes b 3 2. This also requires 1 division step to convert this into 1 by dividing this gets converted into 1, this gets converted into a 2 3 1 or a 2 3 2 and this gets converted into b 2 2.

Where, b 2 2 is nothing but b 2 1 divided by a 2 2 1. So, there is 1 division and to there are 2 division steps exactly, as we had earlier there; is going to be 1 addition subtraction and 1 multiplication division step to get this guy they will be 1 addition subtraction and 1 multiplication division step to get this guy.

So, 2 addition subtraction plus 2 multiplication divisions plus 2 multiplication divisions over here. So, as before we will have 2 addition subtraction and 4 multiplication division steps in step number 2 as well. And we are going to keep doing that for n minus 1 rule just as we did in the Gauss elimination step and that each of these steps is going to require 2 addition subtraction plus 4 multiplication division step.

(Refer Slide Time: 54:53)



So, total computational requirement is going to be n minus 1 multiplied by 2 plus n minus 1 multiplied by 4. These are the addition subtraction steps and these are the multiplication division steps

And this is exactly, where the Thomas algorithm or the tri diagonal matrix algorithm has an advantage over the Gauss elimination that instead of this method is n to the power 3 computational this method having an n to the power 3 computation requirements as n to the power 1 computational requirement.

So, when we go from n equal to ten to n equal to 100 the number of computations required are going to go from order of n; that means, perhaps in going to be 6 multiplied by 9. So, 9 times 6 are 54.

So, it is going to go from 54 computations to 540 computations if we are going to use 100 steps. So, the amount of increase in effort with the increase in total number of n. When we are discretizing; this type of a boundary the amount of increase in effort is manageable when we are going to use a method such as a tri diagonal matrix algorithm.

So, with that I will come to end of this particular lecture 5 of module 3. What we basically, covered in lecture 5 of module 3 is the Gauss elimination and tri diagonal matrix algorithm. And we compared that 2 by looking at amount of computation required for each step and therefore, we looked at the total computational requirement for both the Gauss elimination and the tri diagonal matrix algorithm.