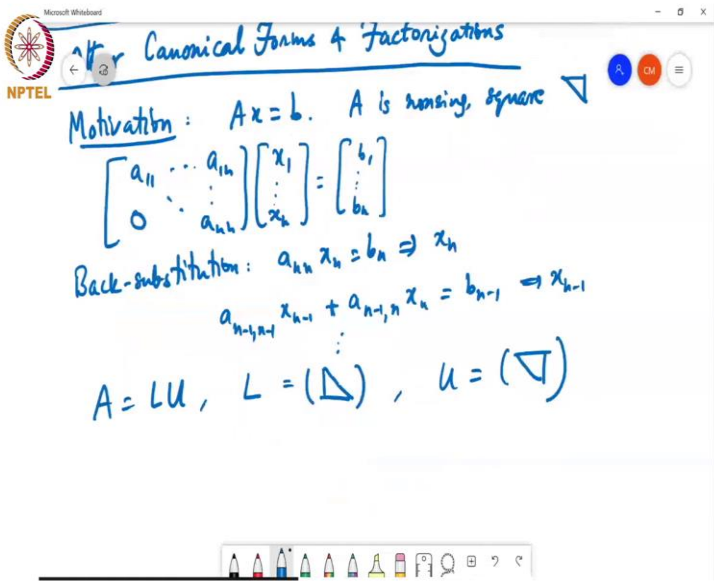
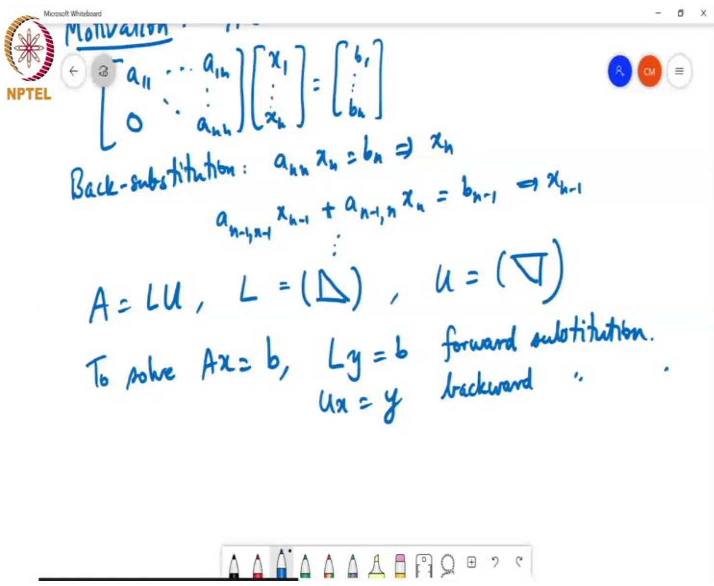
Matrix Theory
Professor Chandra R. Murthy
Department of Electrical Communication Engineering
Indian Institute of Science, Bangalore
Lecture 52
Other Canonical forms and factorisation of Matrices:
Gaussian elimination & LU factorisation

(Refer Slide Time: 00:14)





So, the next thing I want to talk about is we have discussed the Jordan Canonical form so now that gives us a nice opportunity to discuss maybe other canonical forms and factorizations. So, specifically we will look at triangular factorizations where you will reduce the matrix to a triangular form and this is useful because if you are trying to solve a system of linear equations then if A is so suppose so the motivation is that we want to solve Ax equals b and suppose a is non-singular and square and let us say it is upper triangular.

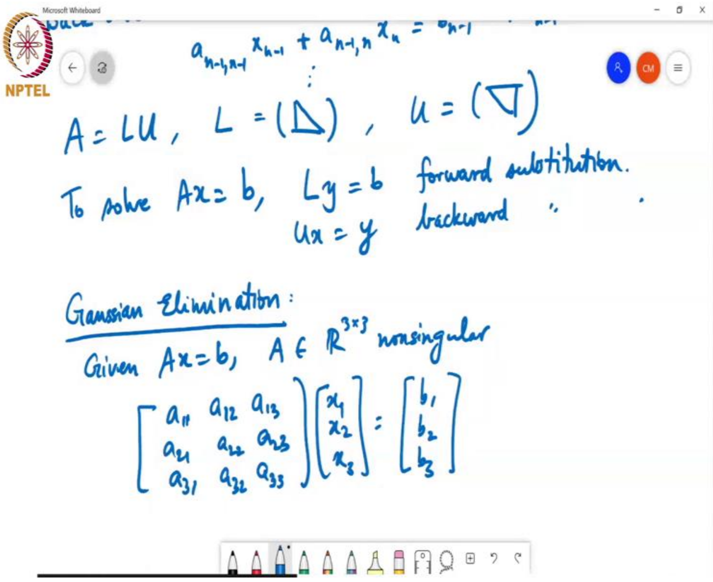
Then what I can do is it is basically this system of equations is of the form a11 to a1n ann 0 times x1 through xn equals b1 to bn. And one way to solve this is through what is known as back substitution we will use the last equation ann xn equals bn and this will directly give us xn and then we substitute that into an minus 1 n minus 1 n minus 1 times xn minus 1 plus an minus 1 n times xn equals bn minus 1 and you can solve you already know what xn is.

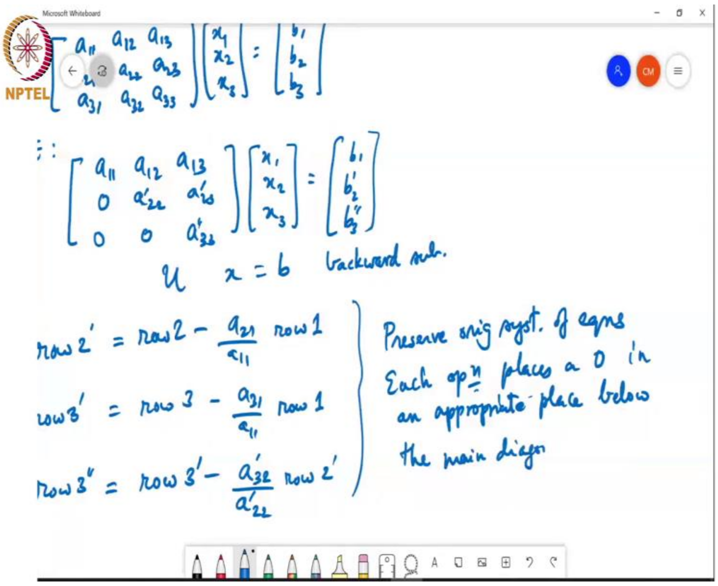
So, you can substitute for xn in here take it to the other side and you can solve for xn minus 1 and so on. So, basically if A is triangular, we can do this but if A is not triangular but it is non-singular but you can almost do what is what I just showed you if we have a factorization that looks like A equal to LU where L is lower triangular and U is a upper triangular. So, then what we can do is ins to solve for   solve for

To solve Ax equals b what we can do is we first solve L let us call it y a is L U so U x I will call it y equals b and L is lower triangular so you can use exactly the opposite of what I discussed here and that is typically also called forward substitution. And then once you found y you you solve U x equals y this is backward as I because U is that upper triangular.

So, if I can find the factorization of a in the form of U then I can solve Ax equals b using these two forward and backward substitution steps. So, this is this is meaningful only if I can compute L and U without too much computational effort. Otherwise, I might as well try to invert a so how do you do this LU factorization? So, the answer to that question lies in what is known as gaussian elimination.

(Refer Slide Time: 04:32)





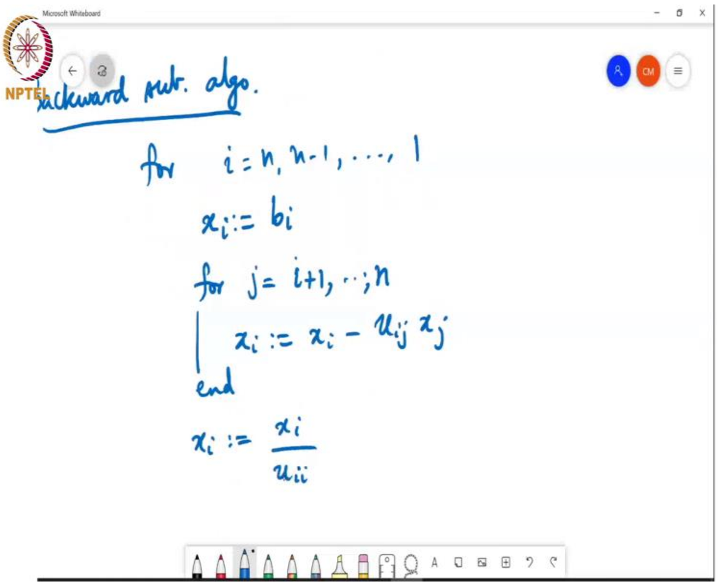
So, we will come to LU factorization in a bit but first this is a small d2 which is a gaussian elimination is one way to solve a system of linear equations I suspect most of you have seen this in your undergraduate already but just to recap. So, suppose just as an example we are given Ax equals b a is some matrix of size 3 cross 3 and it is non-singular. Then basically the system of equations I am trying to solve will look like a11, a12, a13, a21, a22, a23, a31, a32, a33 times x1, x2, x3 is equal to b1, b2, b3.

Then what I can do is I can use gaussian elimination to reduce this to the form a11, a12, a13, 0 a22, dash a23 dash and 0 0 a33 say double dash times x1 x2 x3 is equal to I have to do the same gaussian elimination operations on the right side. So, I will have b1 b2 dash and b3 double dash. And then now I can this is of the form U times x equals b and backward substitution works. So, what are these row operations I do to get this form it is very simple what I have to do is first I compute row 2 with the dash the single dash is equal to row 2 minus a21 over a11 times row 1 to that this element will become 0 and these will give you something else and this b1 dash b2 will become some b2 dash.

And then I do row 3 dash is equal to row 3 minus a31 over a11 times row 1. So, this will kill the bottom right entry of the matrix and then you will have a b3 dash on the right-hand side but these entries may be non-zero and then we compute row 3 double dash equal to row 3 dash minus a32 dash divided by a22 dash times row 2 dash.

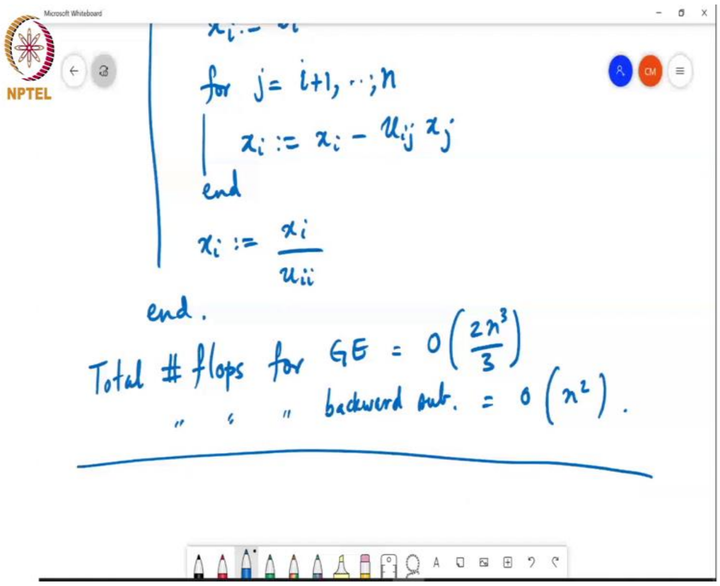
So, if you do these three steps it will reduce the matrix down to this form and then you will have the b2 dash b3 dash and then you can use backward substitution to solve for x. So, basically each row operation it what they do these row operations they preserve the preserve the original system of equations and each operation places 0 in an appropriate place below the main dial. And this is the reason why this gaussian elimination works.

(Refer Slide Time: 08:59)



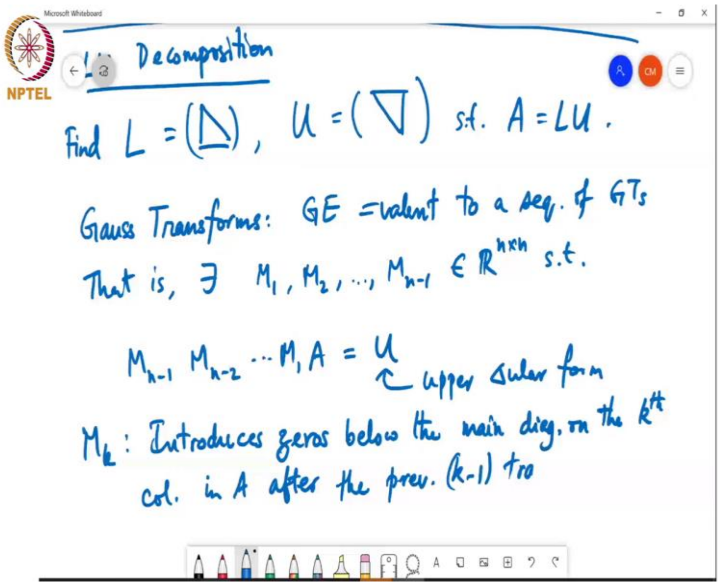
So, basically once a is triangularized we can obtain the solution by backward substitution. So, here is the backward substitution algorithm just for the sake of completeness I have already explained what it is so for i equal to n, n minus 1 down to 1 what you do is you set xi equals bi and then for j equal to i plus 1 to n we set xi, xi minus ui j times xj and then finally you set xi j equals x i over uii.

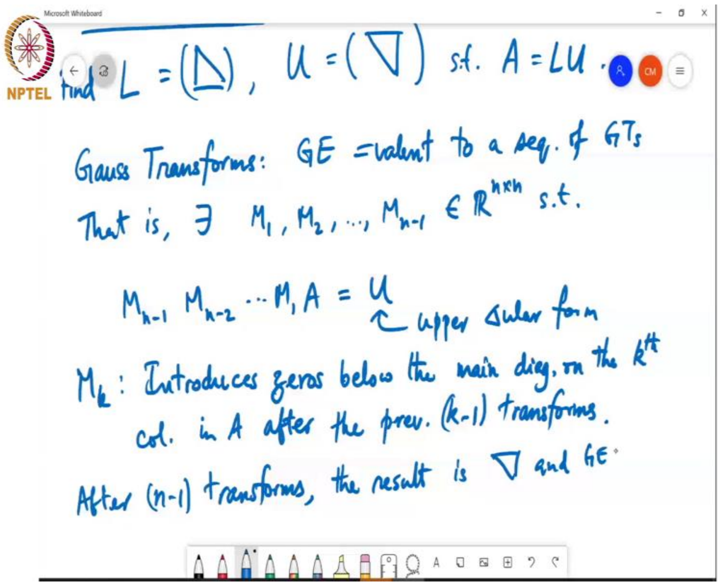
(Refer Slide Time: 10:07)



So, if you if you actually went through the computer effort involved in doing gaussian elimination and backward substitution one counts the number of computational operations in terms of flops or floating-point operations. And so, the total number of floating-point operations for gaussian elimination is of the order of 2n cube over 3 and the total number of flops backward substitution is of the order of n square. So, basically gaussian elimination is the most expensive step in solving for ax equals b via gaussian elimination. So, so with that background we can now discuss about LU decomposition.

(Refer Slide Time: 11:19)





So, we want to find L which is a lower triangular matrix and u which is an upper triangular matrix. Such that A is equal to LU. So, the questions are how do you perform this LU decomposition and what is its computational effort and finally what is the relationship between gaussian elimination and LU decomposition.

So, the first point about finding this or first step in finding this LU decomposition is something transforms. So, basically the gaussian elimination is equivalent to a sequence of gauss transforms what this we will see and we will explicitly write out how this happens that is there exists matrices M1, M2 up to n minus 1 which are in r to the n cross n such that Mn minus 1 Mn minus 2 all the way down to m 1 times A is equal to u and this is in the upper triangular form.

And Mk specifically is a matrix that introduces a zero or zeros below the main diagonal on the kth column after the previous k minus 1 transforms. So, basically after n minus 1 transforms the result is upper triangular and the gaussian elimination is complete. So, the first transform M1 will introduce zeros below the first column of A.

The second M2 M2 M1a will have zeros below the main diagonal of the first two columns of A and so on. So, after kn minus 1 transforms the result is upper triangular and gaussian elimination is complete. So, the thing that we need to understand next is what is the structure of mk so this is something I will discuss in the next class we will stop here for today.