# Elementary Numerical Analysis

**Prof. Rekha P. Kulkarni**
**Department of Mathematics**
**Indian Institute Of Technology, Bombay**

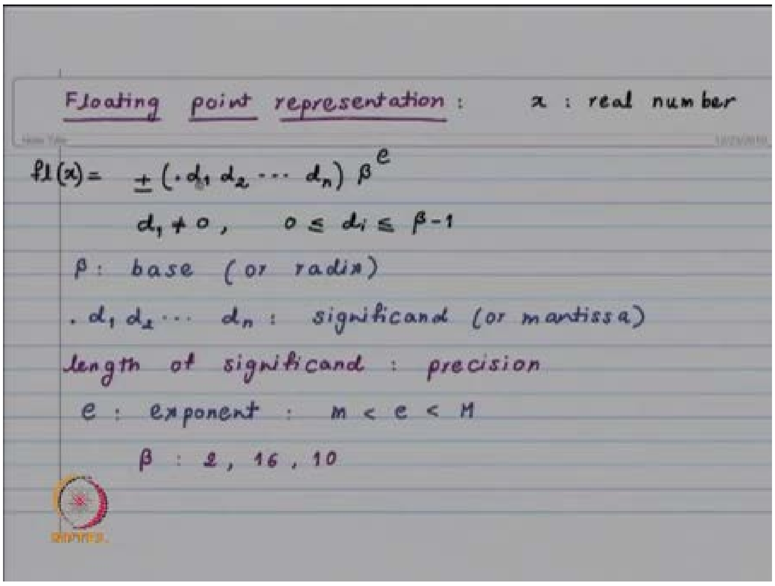
## Lecture No. # 25
## Effect of Small Pivots

In our last lecture we have seen that, if the coefficient matrix a in the system of linear equations a x is equal to b, if it is ill conditioned, that means, if norm of a into norm a inverse if it is a big number, then the solution can be sensitive to the perturbation in the right hand side and in the coefficient matrix. Now, if the matrix a which is given to us, if it is ill conditioned, then we cannot do much about it, but what is in our hands is not to make a well conditioned matrix into ill conditioned matrix by our operations.

Let me explain, if we look at gauss elimination method, in that case we are multiplying a certain row by a non-zero constant and then subtracting from another row; so, if you divide by a small number then your multiplier is going to become big and it can make the originally well-conditioned matrix into an ill conditioned matrix.

So, today we are going to consider these phenomena by an example and then we will consider backward error analysis, we will not go into details, but I want to give you some idea about the backward error analysis. Now, let me recall the floating point representation of a real number and what happens if you subtract two numbers which are about the same.

(Refer Slide Time: 02:31)



Floating point representation : $x$ : real number

$fl(x) = \pm(.d_1 d_2 \cdots d_n)\,\beta^e$

$d_1 \neq 0, \quad 0 \leq d_i \leq \beta - 1$

$\beta$ : base (or radix)

$.d_1 d_2 \cdots d_n$ : significand (or mantissa)

length of significand : precision

$e$ : exponent : $m < e < M$

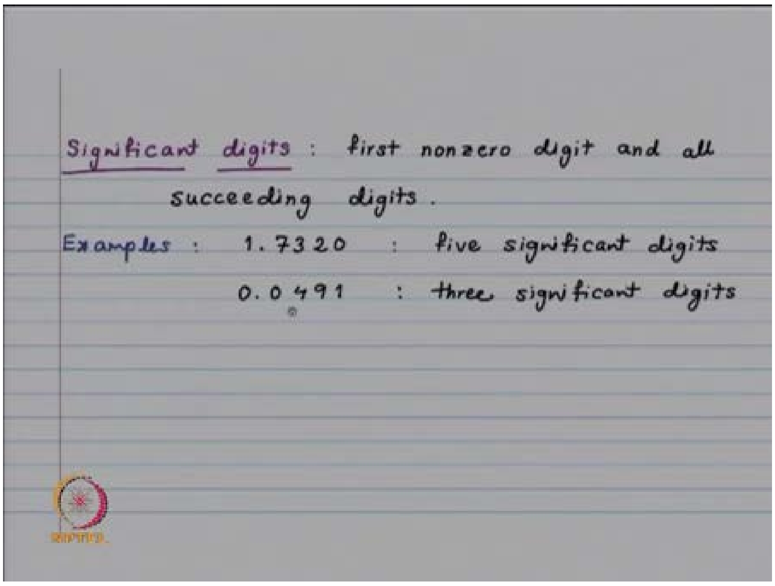
$\beta$ : $2, 16, 10$

So, first the floating point representation of a real number. So, we have got x is a real number, and then floating point of x, it will be, this is the standard form plus or minus then dot d 1 d 2 d n beta raise to e. We will assume that d 1 is not equal to 0, the numbers d i's they are going to lie between 0 and beta minus 1, this beta it is known as base or radix and it takes values binary representation then 2, then decimal representation that means, the basis 10 or hexadecimal then the beta is 16.

The number n that is going to depend on your computer, so in any case these d 1 d 2 d n these are going to be like…, the n is going to be finite and this is known as significant or mantissa. Then e is the exponent; so, the exponent will lie between small m and capital m; so, again the values of small m and capital m, they depend on the computer which you use and whether you are using single precision or double precision.

So, this is floating point representation; if your beta is equal to 2, then your d i's they are going to lie between 0 and 1. So, if we make convention that our representation d 1 is not going to be equal to 0 then we do not need to store it, because then d 1 will be always is equal to 1; so, this is the floating point representation of a real number.

(Refer Slide Time: 04:37)



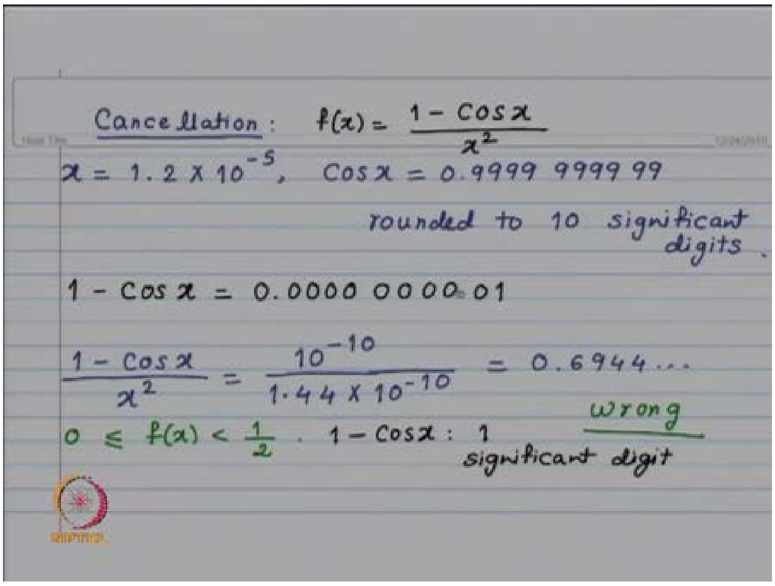
Significant digits : first nonzero digit and all succeeding digits.

Examples : 1.7320 : five significant digits

0.0491 : three significant digits

Now, the definition of significant digits is first non-zero digit and all succeeding digit. So, if you look at the number 1.7320 then there are five significant digits, because first non-zero digit and all succeeding digits, the succeeding digit may be zero or non-zero, whereas if you look at 0.0491 then we have got only three significant digits, then this 0 and this 0 is not counted as significant digit.

(Refer Slide Time: 05:15)



Cancellation : $f(x) = \frac{1 - \cos x}{x^2}$

$x = 1.2 \times 10^{-5}$, $\cos x = 0.9999\ 9999\ 99$

rounded to 10 significant digits.

$1 - \cos x = 0.0000\ 0000\ 01$

$\frac{1 - \cos x}{x^2} = \frac{10^{-10}}{1.44 \times 10^{-10}} = 0.6944\ldots$ wrong

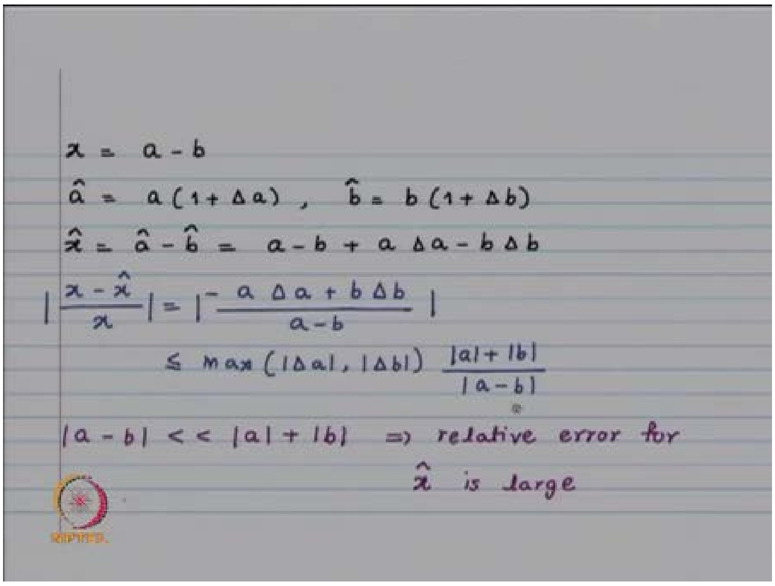
$0 \leq f(x) < \frac{1}{2}$. $1 - \cos x$ : 1 significant digit

Now, let us see what happens in the cancellation; when you subtract two numbers which are above the same, there is going to be loss of (( )); so, we have f x is equal to 1 minus cos x by x square; suppose, your x is 1.2 into10 raise to minus 5 and this is the value of cos x which is rounded to 10 significant digits; then when you look at 1 minus cos x, so we are allowed to have 10 significant digit, when you consider 1 minus cos x then what you get is this number 0 point then 4 0’s 4 0’s 0 and then 1.

In 1 minus cos x there is only one significant digit, so you started with 10 significant digits, and 1 minus cos x you have only one significant digit, so there is a lot of cancellation; then when you look at 1 minus cos x by x square, 1 minus cos x is 10 raise to minus 10 x is 1 point into 10 raise to minus 5, so that is going to be 1.44 into 10 raise to minus 10 and then what you get is 0.6944; now, this number is completely wrong, because what one can show is 0 less than or equal to f x less than half, so this is the catastrophic cancellation; we had number 1 and then another was number which was very near to 1, so when you subtract two numbers which are about the same or whose many digits they coincide, then when you do the subtraction then there is loss of accuracy or loss of significant digits.

Now, what is the way out? So, you had 1 minus cos x upon x square, you can use trigonometric identities and then the same formula you can write it in a different manner. So, what I mean to say is, we have to keep in mind that this phenomena of catastrophic cancellation that as far as possible we should avoid subtracting two numbers which are going to be about equal.

(Refer Slide Time: 07:53)



$$x = a - b$$

$$\hat{a} = a(1 + \Delta a), \quad \hat{b} = b(1 + \Delta b)$$

$$\hat{x} = \hat{a} - \hat{b} = a - b + a\,\Delta a - b\,\Delta b$$

$$\left|\frac{x - \hat{x}}{x}\right| = \left|\frac{-a\,\Delta a + b\,\Delta b}{a - b}\right|$$

$$\leq \max(|\Delta a|, |\Delta b|)\,\frac{|a| + |b|}{|a - b|}$$

$|a - b| << |a| + |b| \Rightarrow$ relative error for $\hat{x}$ is large

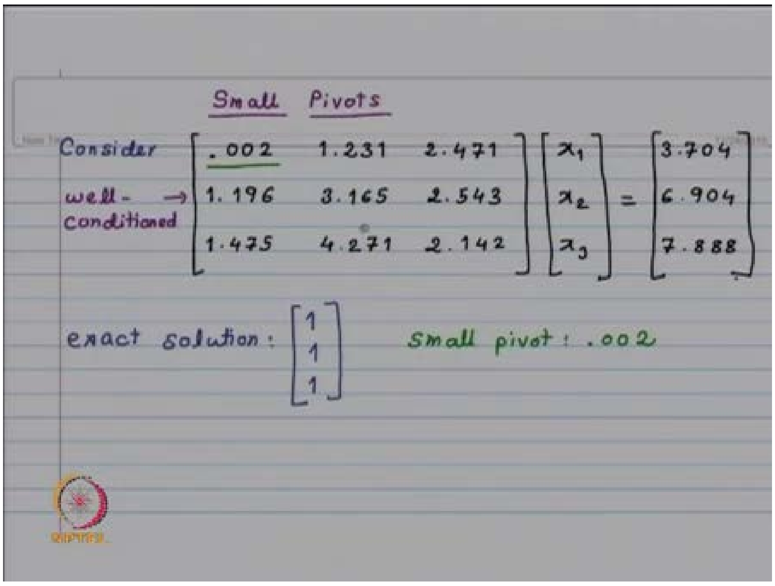
So, here look at the subtraction x is equal to a minus b, so these are the exact values, a cap is the perturbed value, so a cap will be a into 1 plus delta a some error, b cap is going to be b into 1 plus delta b. So, the computed result is going to be x cap is equal to a cap minus b cap which will be equal to a minus b plus a delta a minus b delta b, and hence the relative error will be x minus x cap divided by x which will be minus a delta a plus b delta b divided by a minus b, this is going to be less than or equal to maximum of mod delta a mod delta b into mod a plus mod b divided by modulus of a minus b.

So, in these inequality maximum of modulus of delta a, delta b appearing in you in our error relative error or bound; for relative error that is something normal what you have to focus on is the term mod a plus mod b divided by modulus of a minus b. So, if modulus of a minus b is small, then one upon mod of a minus b will be big, and then you are relative error it can be big; so, once again this illustrates the fact that, if you subtract two numbers which are almost equal then there is going to be a trouble. This is the background.

Now, we are going to look at a system of linear equations in which case the pivot is small. So, if the pivot is small then when you look at the multipliers, they are going to be big, and we are the gauss elimination without partial pivoting then that is going to be affected and the results which we have obtained they will not be reliable or the error is going to be big. So, for the illustration we are going to consider some number of fixed significant digits. So, we are going to do the computations are going to be exact.

But at a time we will be allowed to only keep certain number of digits and this is what happens when you use computer to do your computations that there will be always some fix number of significant digits; then what one does is, one does either rounding of or one does it chopping, but in any case you can work only with finite number of digits.

(Refer Slide Time: 11:07)



Small Pivots

Consider well-conditioned →
$$\begin{bmatrix} .002 & 1.231 & 2.471 \\ 1.196 & 3.165 & 2.543 \\ 1.475 & 4.271 & 2.142 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3.704 \\ 6.904 \\ 7.888 \end{bmatrix}$$

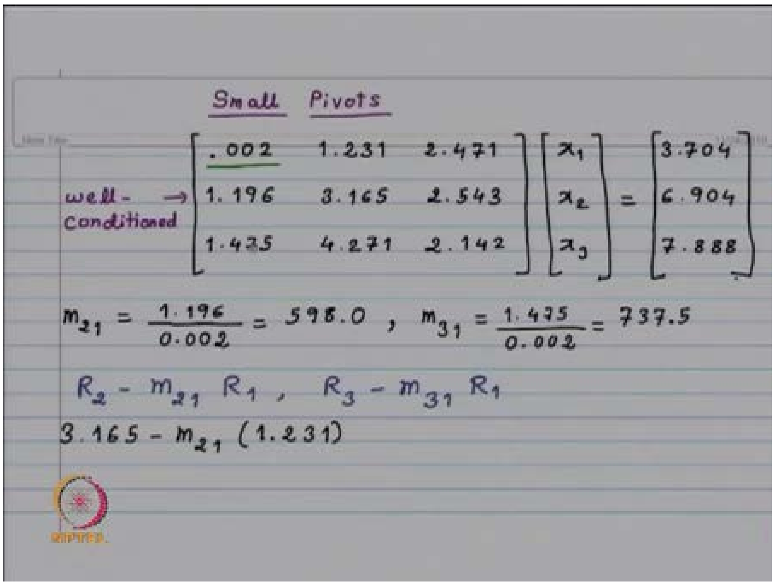
exact solution : $\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ small pivot : .002

So, here is the example, you have got this is a system, and we will be considering four significant digits, that means, at any time you you are allowed to have only four digits; now, here one can verify that it is a well conditioned matrix.

This is matrix a, one can calculate a inverse and then one can look at either infinity norm or one norm, and one can check that the matrix is well conditioned; the right hand side is so chosen that the exact solution is 1 1 1; if you look at this 0.002, this is a small pivot, but it is still not equal to 0. So, I will be perfectly justified in choosing this as my pivoted entry. So, in the gauss elimination method what we are going to do is, we are going to introduce 0's in the first column below the diagonal, that we will do by subtracting, say a to 1 by a 1 1 times first row from the second row, and a three 1 by a 1 1 times first row from the third row; and here are a 1 a 1 1, that is a small number, so that is why a 2 1 by a 1 1 that will be a big number.

So, we will be subtracting large multiples of the first row from the second row and from the third row; so, when you do this in the process your next sub matrix we on which you are going to work that will become ill condition.

(Refer Slide Time: 13:11)



Small Pivots

$$\text{well-conditioned} \rightarrow \begin{bmatrix} .002 & 1.231 & 2.471 \\ 1.196 & 3.165 & 2.543 \\ 1.475 & 4.271 & 2.142 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3.704 \\ 6.904 \\ 7.888 \end{bmatrix}$$

$$m_{21} = \frac{1.196}{0.002} = 598.0 \; , \; m_{31} = \frac{1.475}{0.002} = 737.5$$

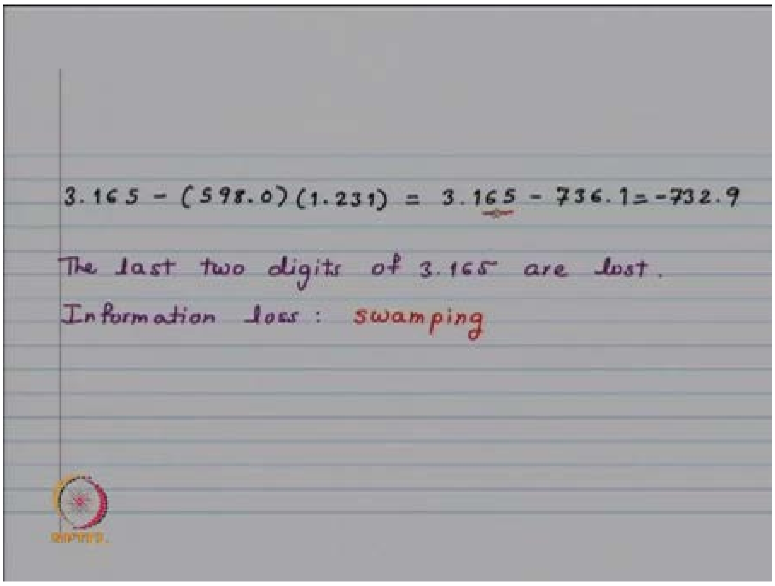
$$R_2 - m_{21} R_1 \; , \; R_3 - m_{31} R_1$$

$$3.165 - m_{21} (1.231)$$

So, let us see our m 2 1 is 1.196 divided by 0.002 it is a 2 1 by a 1 1; so, it is 598.0; note that this 0 is significant, it is at a time you clear around 2 retain 4 digits m 3 1 will be a 3 1 by a 1 1.

So, it is again 1.475 divided by 0.002, so it is 737.5; the operations which we are going to do are R 2 minus m 2 1 R 1 and R 3 minus m 3 1 R 1, these operations will introduce 0’s here, so this is the first step of gauss elimination method; so, here we know that we are going to get 0; so, the first thing we will be doing is 3.165 minus m 2 1 times this 1.231. So, let us do this subtraction, the way we are going to do the subtraction multiplication is at any stage retain only four digits.

(Refer Slide Time: 14:26)



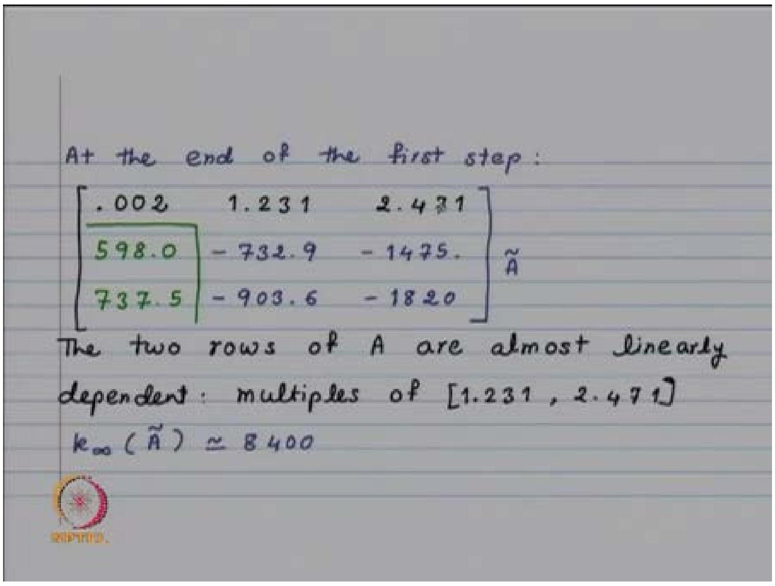
$3.165 - (598.0)(1.231) = 3.\underline{165} - 736.1 = -732.9$

The last two digits of 3.165 are lost.

Information loss : swamping

So, here is the result, 3.165 minus 598.0 into 1.231 that will give us 3.165 minus 736.1; so, here when you multiply, then retain only 4 digits, so you get this 736.1; when I do the subtraction again I am allow to retain only four digits; so, I will get minus 732.9; that means, these 2 digits 65 which were significant, these are lost, and this information loss it is known as swamping; so, this was for the element a 2 2 and same thing is going to happen for the other elements.

(Refer Slide Time: 15:20)



At the end of the first step:

$$\begin{bmatrix} .002 & 1.231 & 2.431 \\ 598.0 & -732.9 & -1475. \\ 737.5 & -903.6 & -1820 \end{bmatrix} \tilde{A}$$

The two rows of A are almost linearly dependent: multiples of [1.231, 2.471]

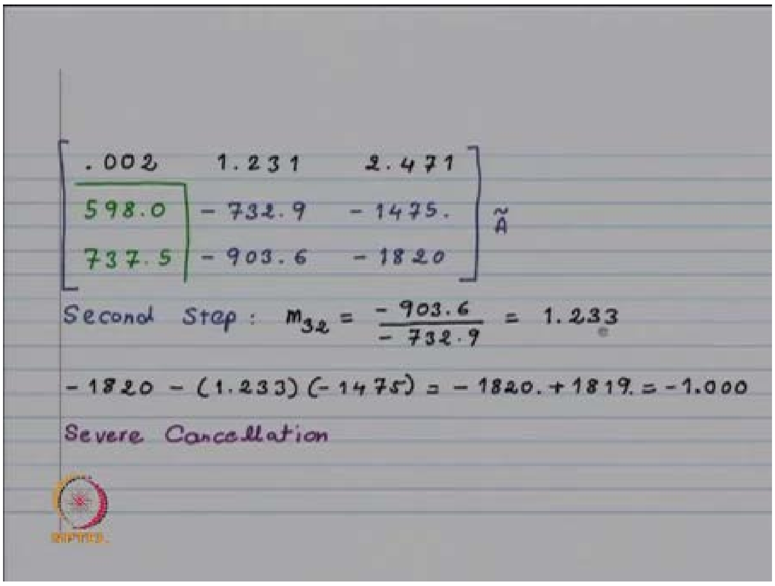
$k_\infty(\tilde{A}) \simeq 8400$

And at the end of the first step what we get is the first row is unaffected, here we have zeros, so that is where I am writing our multiplier and then you get minus 732.9 minus 1475 minus 903.6 and minus 1820.

So, in our next step we are going to work on this matrix. So, we will be subtracting appropriate multiple of the second row from the third row; look at this 2 by 2 matrix, this 2 by 2 matrix its 2 rows are almost linearly dependent, because they are almost multiples of 1.231 and 2.471, so this condition number of A tilde with infinity norm is going to be big, it is about 8400; if you remember we had seen that the ill conditioning of the matrix it has nothing to do with the small determinant, but it has to do something with linear independence and linear dependent. So, we started with A matrix A which was well conditioned, because of our small pivot the multipliers became big and then we subtracted large multiples of the first row from the second row and the third row. So, in the process the modified 2 by 2 matrix, its rows they were almost multiples of the same vectors, so they become almost linearly dependent, and that is what makes the condition number of that matrix to be big; now, if the condition number is big, then it is going to be sensitive to the perturbation.

And then the solution which which we get the computed solution it will be much further away from the exact solution. So, we have seen the first step of gauss elimination method.

(Refer Slide Time: 17:47)



$$\begin{bmatrix} .002 & 1.231 & 2.471 \\ 598.0 & -732.9 & -1475. \\ 737.5 & -903.6 & -1820 \end{bmatrix} \tilde{A}$$

Second Step : $m_{32} = \dfrac{-903.6}{-732.9} = 1.233$

$-1820 - (1.233)(-1475) = -1820. + 1819. = -1.000$

Severe Cancellation

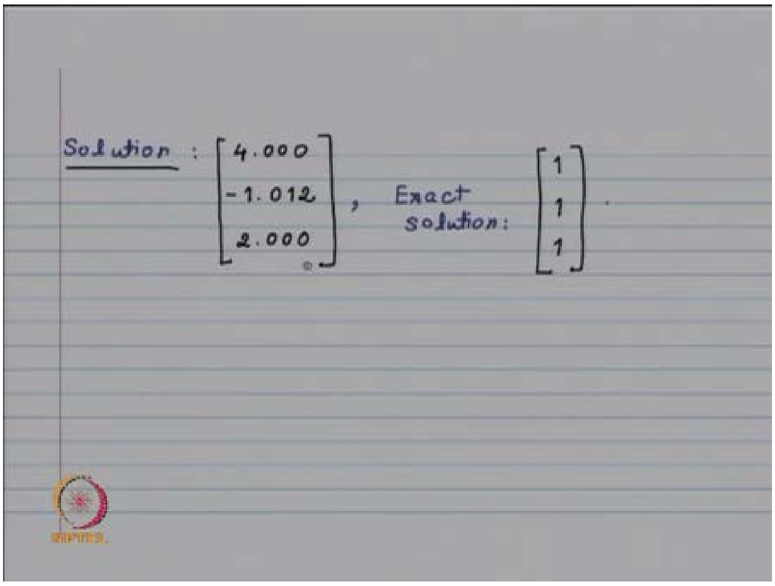
So, let us continue. So, in the second step, our multiplier will be 903.6 divided by 732.9, so it is 1.233 the multiplier is not big and then here any way we are going to get 0.

So, we need to do the subtraction minus 1820 minus multiply minus 1475 by 1.233, when you multiply you are going to get minus 1820 plus 1819; now, in this subtraction again there is severe cancellation, there is catastrophic cancellation, so you are losing a lot of significant digits and then you have reduce the system a x is equal to b to u x is equal to y, the swamping and severe cancellation that will occur also in the back substitution; you can verify that these are the values you get, only thing is when you do the computations even though you are doing hand computation.

Whatever result you get like you multiply you multiply you get some 6 7 digits, so then you truncate or you round it off and at every stage keep only 4 digits, that is how you are going to do the computations; if for this system, if you assume that you have got infinitely many digits at your disposal the way we do for hand computation then there will not be a problem.

The problem is because you can retain only four digits at a time and that is what happens when you are doing computations using computer. So, we have got a x is equal to b, we reduce it to upper triangular form, we do the back substitution, and then the solution which you get the exact solution was 1 1 1 and solution which you obtain is completely different.

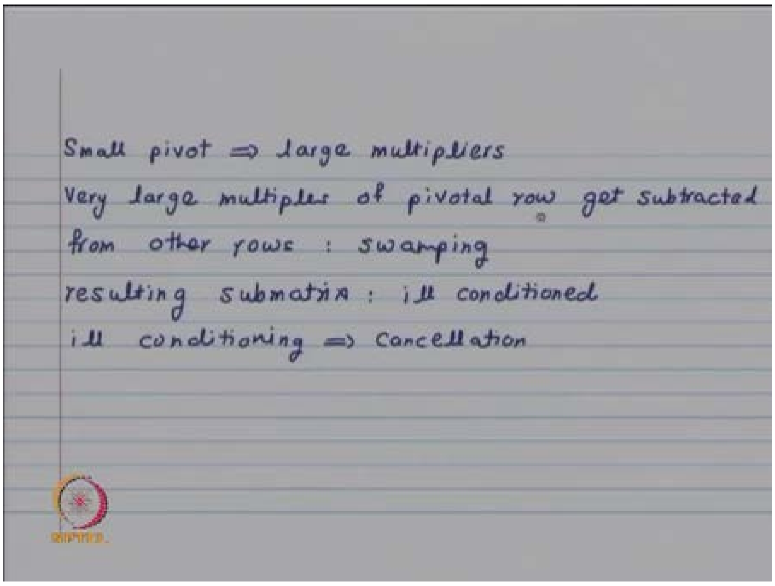
(Refer Slide Time: 20:02)



Solution : $\begin{bmatrix} 4.000 \\ -1.012 \\ 2.000 \end{bmatrix}$, Exact solution: $\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$.

It has got the first component to be 4, second component to be minus 1.012, and the third component to be 2.000.

(Refer Slide Time: 20:12)



Small pivot $\Rightarrow$ large multipliers

Very large multiples of pivotal row get subtracted from other rows : swamping

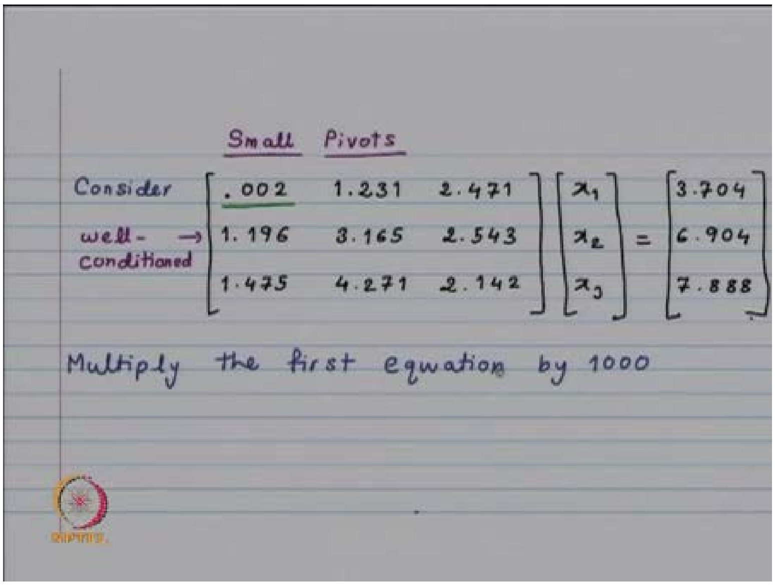
resulting submatrix : ill conditioned

ill conditioning $\Rightarrow$ Cancellation

So, what had happened was, small pivot it implies large multipliers, so very large multipliers of the pivotal row get subtracted from other rows, so there is loss of information that is swamping resulting sub matrix is ill conditioned and ill conditioning it leads to cancellation.

So, these where the problems; the main problem in this whatever we have done, it was the small pivot, the multiplier was big, thus pivot was small, it was 0.002, why not do the multiply the first equation by say thousand, if I do that then the pivot will become 2 and then there should not be any problem; when I consider a system of linear equations and multiply a equation by a non-zero constant, I do not change the system, the solution of the new system is going to be exactly same as the original system. So, let us see whether this will work, that the problem was that 0.002, the pivot was small, so let me multiply by 1000 the first row and see what happened.

(Refer Slide Time: 21:44)



Small Pivots

Consider

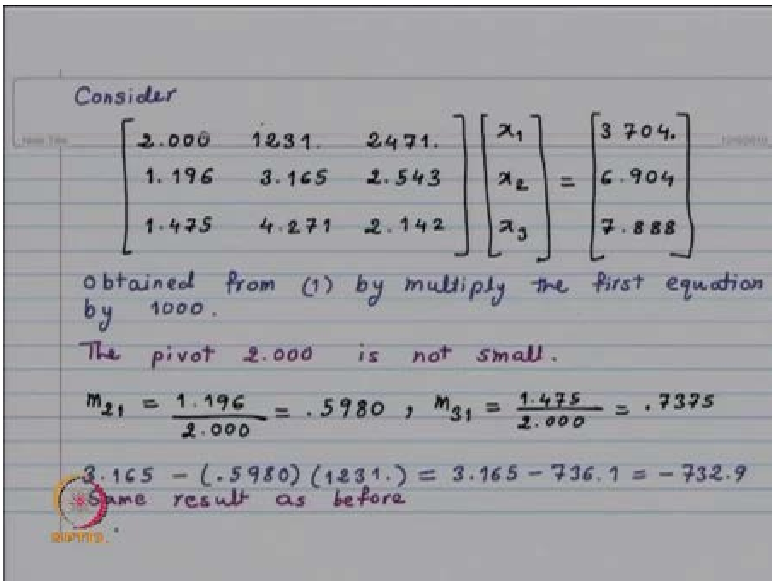
well-conditioned →

$$\begin{bmatrix} .002 & 1.231 & 2.471 \\ 1.196 & 3.165 & 2.543 \\ 1.475 & 4.271 & 2.142 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3.704 \\ 6.904 \\ 7.888 \end{bmatrix}$$

Multiply the first equation by 1000

So, this was our original system and then you multiply the first equation by 1000; so, I am throughout I am going to multiply; so, only the first equation will change.

(Refer Slide Time: 21:46)



Consider

$$\begin{bmatrix} 2.000 & 1231. & 2471. \\ 1.196 & 3.165 & 2.543 \\ 1.475 & 4.271 & 2.142 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3704. \\ 6.904 \\ 7.888 \end{bmatrix}$$

obtained from (1) by multiply the first equation by 1000.

The pivot 2.000 is not small.

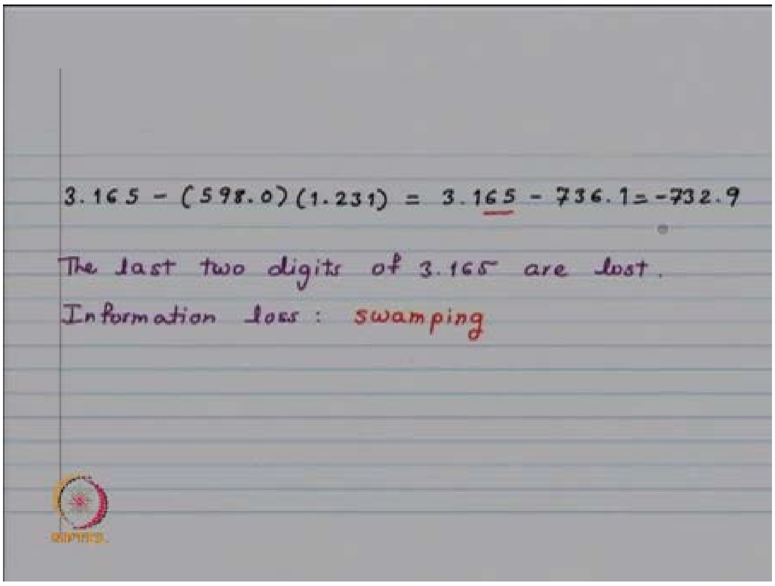
$m_{21} = \frac{1.196}{2.000} = .5980$ , $m_{31} = \frac{1.475}{2.000} = .7375$

$3.165 - (.5980)(1231.) = 3.165 - 736.1 = -732.9$

Same result as before

So, after multiplication this is going to be my new system; the second and third row they are the same, the pivot 2.00 is not small; now, look at m 2 1, so it is 1.196 divided by 2, so it is 0.5980; the m 3 1 which is a 3 1 by a 1 1, that is 0.7375; when I consider 3.165 minus the pivot 0.5980 into 1231, when I do this I get exactly the same number as before.

(Refer Slide Time: 22:40)



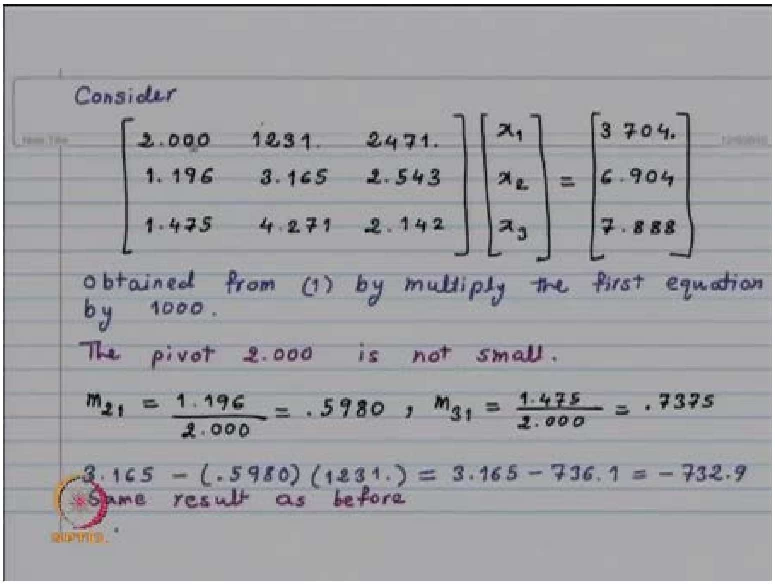
$3.165 - (598.0)(1.231) = 3.1\underline{65} - 736.1 = -732.9$

The last two digits of 3.165 are lost.

Information loss : swamping

Let us see, we had here, when we had considered we got minus 732.9.

(Refer Slide Time: 22:47)



Consider

$$\begin{bmatrix} 2.000 & 1231. & 2471. \\ 1.196 & 3.165 & 2.543 \\ 1.475 & 4.271 & 2.142 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3704. \\ 6.904 \\ 7.888 \end{bmatrix}$$

obtained from (1) by multiply the first equation by 1000.

The pivot 2.000 is not small.

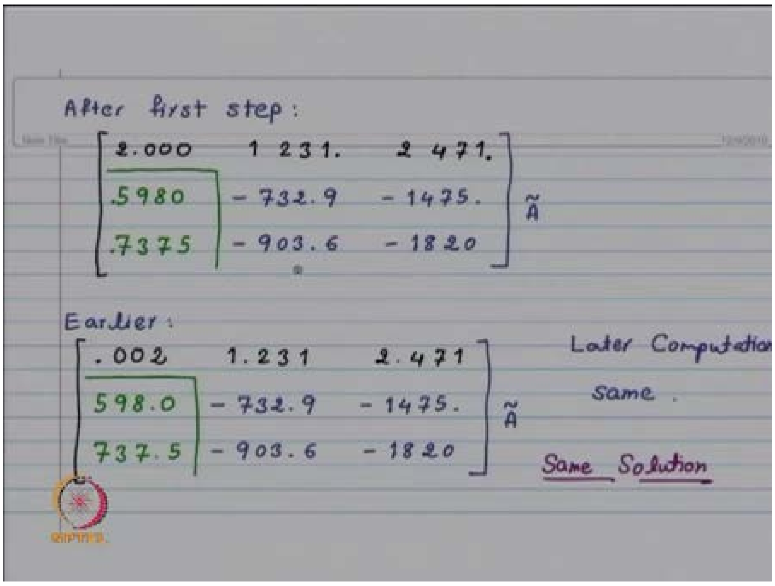
$m_{21} = \frac{1.196}{2.000} = .5980$ , $m_{31} = \frac{1.475}{2.000} = .7375$

$3.165 - (.5980)(1231.) = 3.165 - 736.1 = -732.9$

Same result as before

And now, when we do the new calculations again you are going to get minus 732.9, so this was for this particular element.

(Refer Slide Time: 23:02)



After first step :

$$\left[\begin{array}{c|cc} 2.000 & 1\,231. & 2\,471. \\ \hline .5980 & -732.9 & -1475. \\ .7375 & -903.6 & -1820 \end{array}\right] \tilde{A}$$

Earlier :

$$\left[\begin{array}{c|cc} .002 & 1.231 & 2.471 \\ \hline 598.0 & -732.9 & -1475. \\ 737.5 & -903.6 & -1820 \end{array}\right] \tilde{A}$$

Later Computation same .

Same Solution

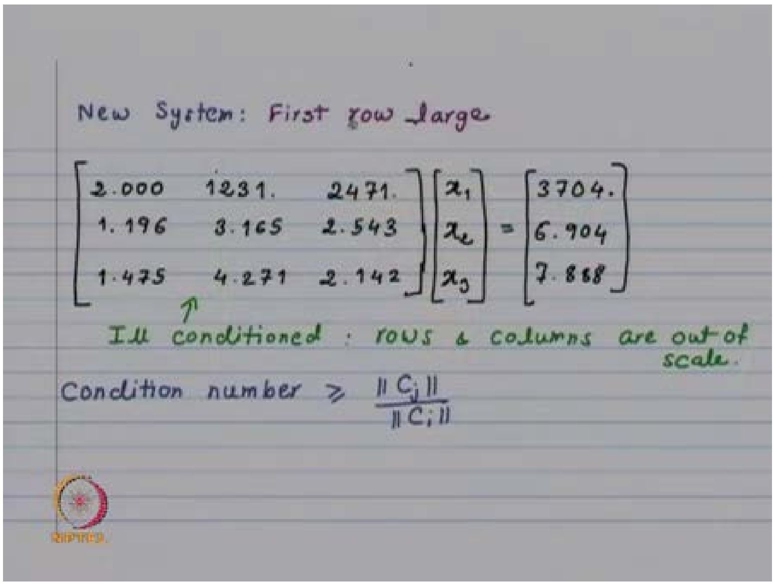
Now, let us see how the other element they get modified in the other element you are going to have exactly the same thing like this was our new system after multiplying by the 1000 the pivots are the multipliers they are small the 2 by 2 matrix here it is exactly same as before.

So, this was in the earlier case this was our first row, the pivots were big, but a tilde 2 by 2 matrix is going to be the same, so later computations they will be the same, and then you are going to get exactly the same solution as before. So, what happened? What we did was we multiplied the first row by 1000 and made the pivot element to be say it was sort of arbitrary that we made to be 2; but recall our result that, if you multiply a matrix a by a non-zero number, then the condition number remains exactly the same; because if I consider instead of a say alpha times a, then I have to look at inverse of alpha times A, that will be one up on alpha times A inverse, so the condition number remains exactly the same; but if I multiply a row by a non-zero number then the condition number is going to be becoming…, it will change, that was the idea in rows scaling and columns scaling; we have seen that, if you have got 2 columns which are like 1 column it has got norm much bigger than the norm of the other column then your condition number is going to be big.

So, one tries to do the scaling, so that rows and columns they are as far as possible of the same order; now, here we had our matrix to be well conditioned; in order to get rid of small pivot we multiplied the first row by a big number, in the process the well conditioned matrix we make it ill conditioned.

So, that is the problem. So, this sort of thing making the pivot arbitrarily large is not going to work. So, here in this case the solution is…, do not do gauss elimination without pivoting, but use it with partial pivoting; that means, look at the first column, look at the entry which has got maximum modulus, interchange the corresponding rows, and then the results which you get they will be acceptable.

(Refer Slide Time: 26:14)



New System: First row large

$$\begin{bmatrix} 2.000 & 1231. & 2471. \\ 1.196 & 3.165 & 2.543 \\ 1.475 & 4.271 & 2.142 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3704. \\ 6.904 \\ 7.868 \end{bmatrix}$$

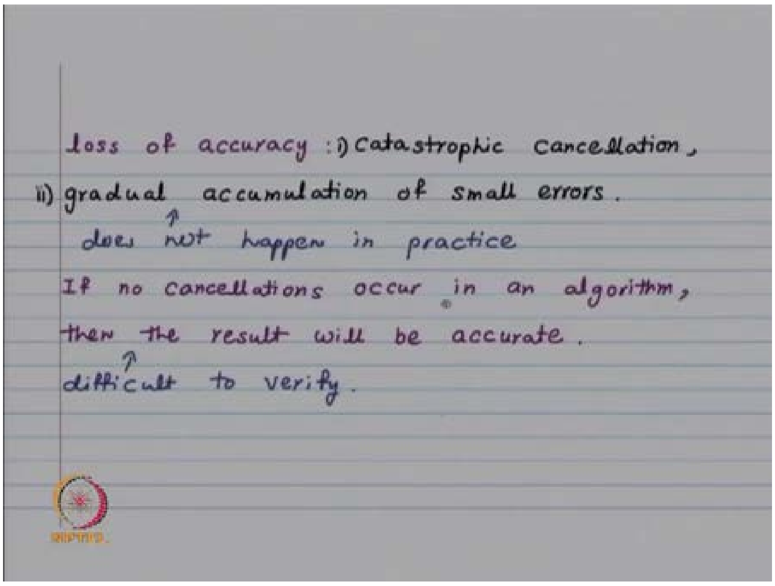
↑

Ill conditioned : rows & columns are out of scale.

Condition number $\geq \dfrac{\|C_j\|}{\|C_i\|}$

So, here as I said earlier system we had small pivot and large multiplier; in the new system the first row is large; now, if the first row is large, see you here you have got 1231, 2471. So, the rows and columns are out of scale, we had seen that condition number is bigger than or equal to norm c j by norm c I, where these are the jth column and ith column and what one says about the columns it is true for rows also. So, this system is ill conditioned and that was the problem.

(Refer Slide Time: 26:54)



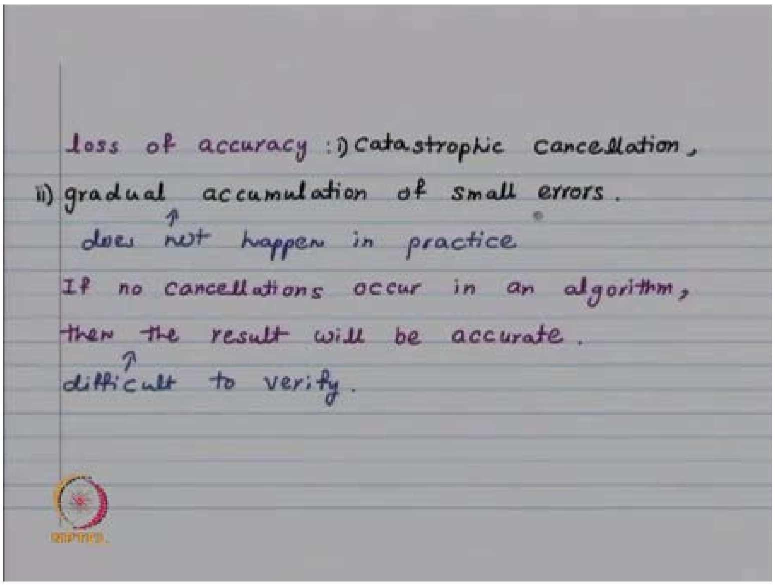
loss of accuracy : i) Catastrophic cancellation,
ii) gradual accumulation of small errors.
does not happen in practice
If no cancellations occur in an algorithm,
then the result will be accurate.
difficult to verify.

So, now, let us look at the error analysis; we have to start with a system a x is equal to b, then when we are using computer instead of system a x is equal to b we are going to solve nearby system; now, we have to see whatever computations we do, we are going to do in finite precision; so, there are going to be two sorts of error, one will be because of the catastrophic cancellation and another will be at each stage there is round of error and then it will keep accumulating. So, these are the 2 errors.

Now, the round off errors, accumulation of that does not happen much in practice. So, if I want to look at the gauss elimination method with pivoting or without pivoting; then in the forward error analysis what one does is, at each stage see we are going to perform various operations, we are going to subtract multiple of a row from the another row, then we will be doing back substitution forward substitution.

So, in each operation if I can guarantee that there is no catastrophic cancellation, then we can say that the error is going to be accepted; now, this is something very difficult that keeping you know track of all the errors at every stage. So, then instead of that what one does is one does backward error analysis.

(Refer Slide Time: 28:52)



loss of accuracy : i) Catastrophic cancellation,

ii) gradual accumulation of small errors.

does not happen in practice

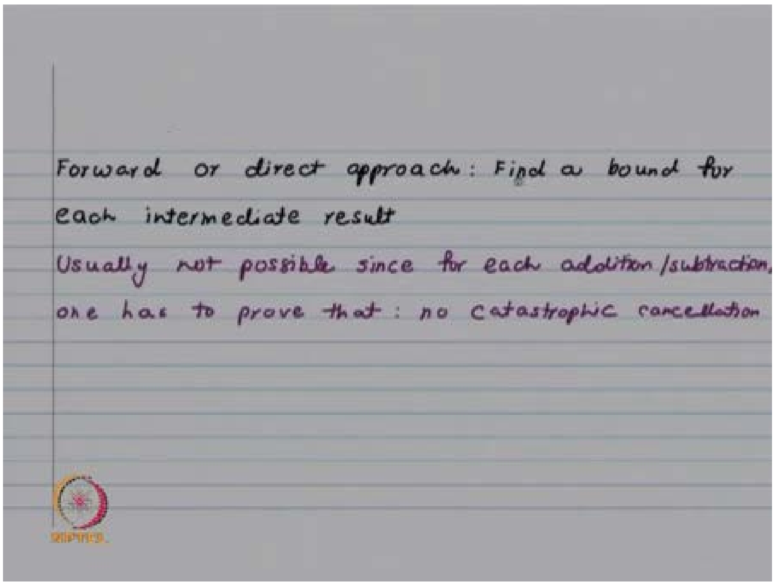
If no cancellations occur in an algorithm,

then the result will be accurate.

difficult to verify.

You have gradual accumulation of small errors, which does not happened in practice; if no cancellation occur in an algorithm then the result will be accurate or accurate enough this is something difficult to verify.

(Refer Slide Time: 29:07)

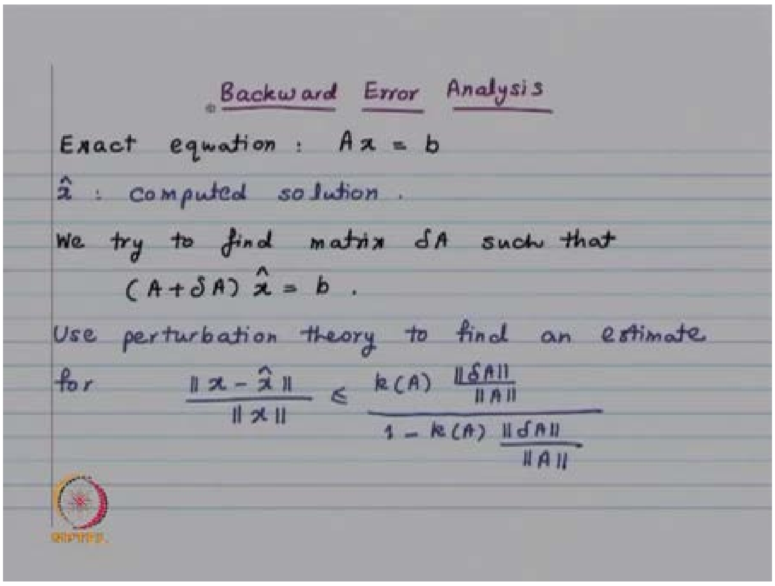


Forward or direct approach : Find a bound for each intermediate result

Usually not possible since for each addition/subtraction, one has to prove that : no catastrophic cancellation

In the forward or direct approach one finds a bound for each intermediate result and this is something not possible, because for each addition or subtraction one has to prove that there is no catastrophic cancellation.

(Refer Slide Time: 29:24)



Backward Error Analysis

Exact equation : $Ax = b$

$\hat{x}$ : computed solution.

We try to find matrix $\delta A$ such that

$(A+\delta A)\hat{x} = b$.

Use perturbation theory to find an estimate

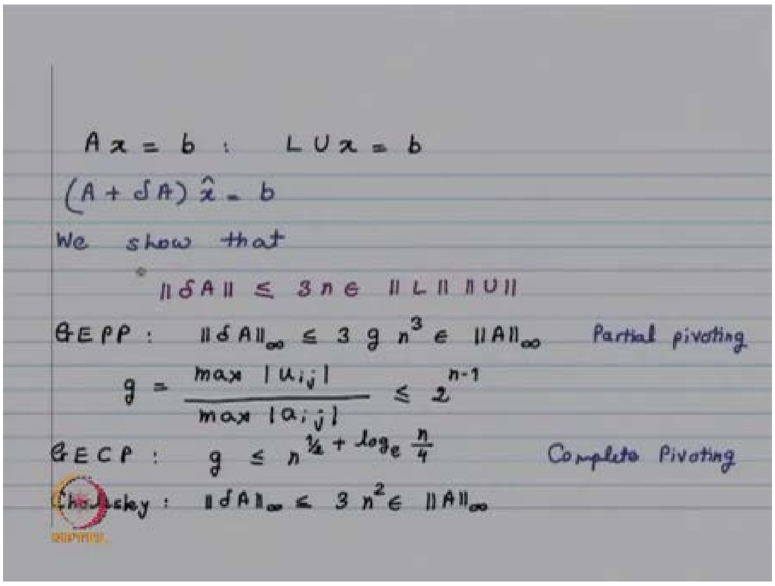
for $$\frac{\|x-\hat{x}\|}{\|x\|} \leq \frac{k(A)\frac{\|\delta A\|}{\|A\|}}{1-k(A)\frac{\|\delta A\|}{\|A\|}}$$

So, now, about the backward error analysis. So, you have exact equation A x is equal to b, x cap is a computed solution, so one tries to find A matrix delta a, such that, A plus delta A x cap is equal to b; and then use perturbation theory to find an bound for relative error norm of x minus x cap by a norm x is less than or equal to we have seen that condition number of a norm delta A by A norm upon 1 minus condition number of a norm delta a by norm a.

we have system of equations A x is equal to b; you are either going to do cholesky decomposition or gauss elimination with partial pivoting or without pivoting and we get a computed solution x cap; if we can show that this computed solution x cap is exact solution of a nearby system, that means, A plus delta A x cap is equal to b; so, if I can find such a delta a then I can use the perturbation theory which we have developed to say something about norm of x minus x cap by norm A; the condition number of a it appears in that bound, that is going to be something inheritant to the system.

So, we cannot do much about that another term which is coming in the bound is norm delta A by norm A. So, if our computed solution x cap is the exact solution of a nearby system with norm delta A to be small, then ours relative error is going be something small; now, this is backward error analysis; we will not be doing this in detail, I just want to state some results, so what is possible.

(Refer Slide Time: 31:51)



$A x = b \ : \quad L U x = b$

$(A + \delta A)\hat{x} = b$

We show that

$\|\delta A\| \leq 3 n \epsilon \ \|L\| \ \|U\|$

GEPP : $\|\delta A\|_\infty \leq 3 \, g \, n^3 \epsilon \ \|A\|_\infty$ Partial pivoting

$$g = \frac{\max |u_{ij}|}{\max |a_{ij}|} \leq 2^{n-1}$$

GECP : $g \leq n^{\frac{1}{2} + \log_e \frac{n}{4}}$ Complete Pivoting

Cholesky : $\|\delta A\|_\infty \leq 3 n^2 \epsilon \ \|A\|_\infty$

Like look at A x is equal to b and then look at L U decomposition of the matrix L U x is equal to b; then what one can show is, norm delta A is less than or equal to three times n epsilon norm L norm U, the delta A norm it is going to depend on the norm of L and norm of U; if you are doing gauss elimination with partial pivoting, then the entry of norm L they are going to be have modulus less than or equal to 1, so then this will be something acceptable; so, then we have to see how norm u increases.

In case of gauss elimination with partial pivoting, what one can show is norm delta A infinity to be less than or equal to 3 g n cube epsilon norm A infinity; epsilon is going to be precision of your computer; n is the size of the matrix; and then g it is known as the growth factor. So, this g is less than or equal to 2 raise to n minus 1 and one can construct an example where g is equal to 2 raise to n minus 1. And 2 raise to n minus 1 is going to grow much faster than n cube and that is something one has to worry about; if you consider gauss elimination with complete pivoting, then your g the growth factor is going to grow much slower; and in case of cholesky decomposition there is no growth factor. So, let me summarize.

If you consider cholesky decomposition then there is no growth factor and that is why the method is going to be stable. So, that was the reason that, of course, you cannot do cholesky decomposition for all systems, your matrix it should be a positive definite matrix then only you can do the cholesky decomposition, but still when it is possible then it is going to be a stable method.

When you consider gauss elimination with complete pivoting, so in the complete pivoting what one does is, look at all the elements of your matrix, there are n square; look at the one which has got maximum modulus and interchange row and columns so that this element of the maximum modulus it occupies the space 1 1, that means, first row first column and then do similarly.

So, this is gauss elimination with complete pivoting and then in this case also the growth factor does not increase too fast and the method will be stable; however, this complete pivoting is going to be expansive, because you need to do a lot of comparisons when you consider gauss elimination with partial pivoting; in case of partial pivoting your growth factor, as I said you can construct examples when the growth factor is 2 raise to n minus 1, so that can be a very big number. So, but in practice like when people have done extensive computations, it was realized that it does not happen in practice.

So, gauss elimination method with partial pivoting it really works well; gauss elimination method without pivoting you should not do, because we have seen that what can happen is you are starting matrix is well conditioned and then it can become ill conditioned. So, among the methods which we have studied, if your matrix is positive definite use cholesky decomposition; if your matrix is not positive definite then one sort of compromises and one uses gauss elimination with partial pivoting. So, this completes our study about solution of system of linear equations, these where the direct methods; now, there are methods which are known as the indirect methods or the iterative methods.

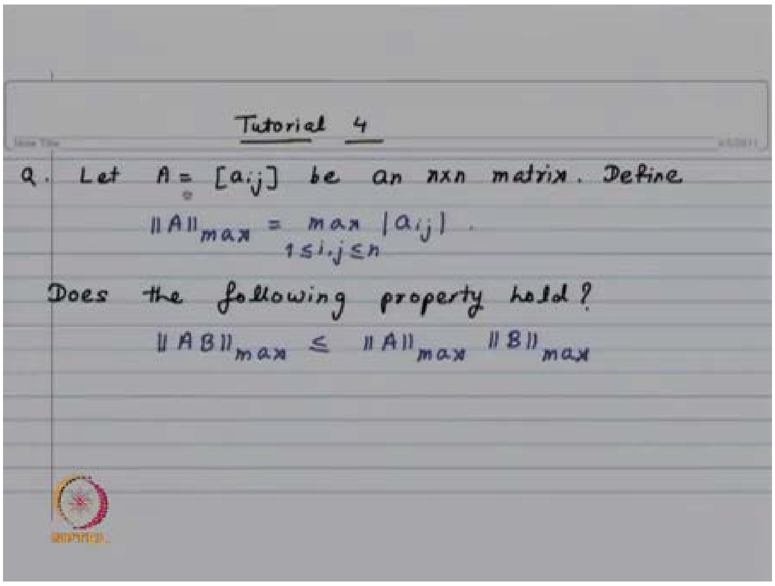
So, we will be considering those methods, but before we do those methods we are going to look at some of the problems and we will first consider solution of non-linear equations. So, that means, if you want to find 0 of a function, so f x is equal to 0, so we will be first considering those that topic and then at the end we will consider 2 iterative methods which are known as Jacobi method and gauss seidel method.

So, let us look at some of the problems. We have considered vector norms and matrix norms; so, for the vector norm we had defined 1 norm, infinity norm, and the 2 norm, then you can define analog of these norms for the matrices; so, analog of the 2 norm or Euclidian norms for the vector is the frobenius norm.

Now, what we said was that, instead of defining these we will define what are known as induced matrix norm; so, start with a vector norm and then you define norm a to be maximum of norm a x by norm x x not equal to 0 vector; this induced matrix norm it has got some desirable property, we have got the fundamental inequality, norm a x is less than or equal to norm A into norm x, where A is matrix x is a vector; we also have consistency condition.

You can multiply two matrices provided they are of appropriate size. So, if A and B are two square matrices I can multiply them; so, we have got consistency condition norm A B less than or equal to norm A into norm B; now, I want to consider analog of maximum norm for the vector A - it is a analog for the matrix norm - and show that it does not satisfy the consistency condition.

(Refer Slide Time: 39:55)



Tutorial 4

Q. Let $A = [a_{ij}]$ be an $n \times n$ matrix. Define

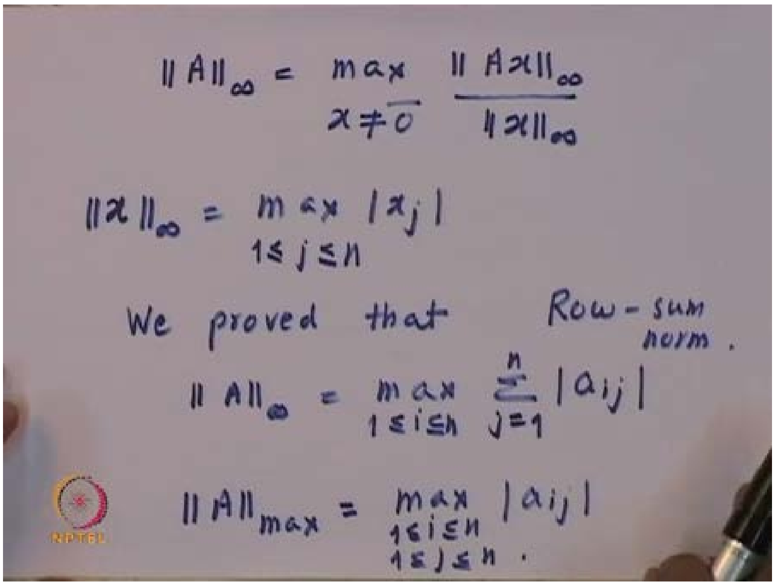
$$\|A\|_{max} = \max_{1 \le i,j \le n} |a_{ij}| .$$

Does the following property hold?

$$\|AB\|_{max} \le \|A\|_{max} \, \|B\|_{max}$$

So, here is the example or a problem, a is n by n matrix; suppose I define norm a max is equal to maximum of modulus of a i j one less than or equal to i j less than or equal to n. So, does the following properties hold that norm AB max is less than or equal to norm A max norm B max. Now, I have not used the notation norm a infinity, because we have reserved that notation for the induced matrix norm.

(Refer Slide Time: 40:40)



$$\|A\|_\infty = \max_{x \neq \vec{0}} \frac{\|Ax\|_\infty}{\|x\|_\infty}$$

$$\|x\|_\infty = \max_{1 \leq j \leq n} |x_j|$$

We proved that Row-sum norm.

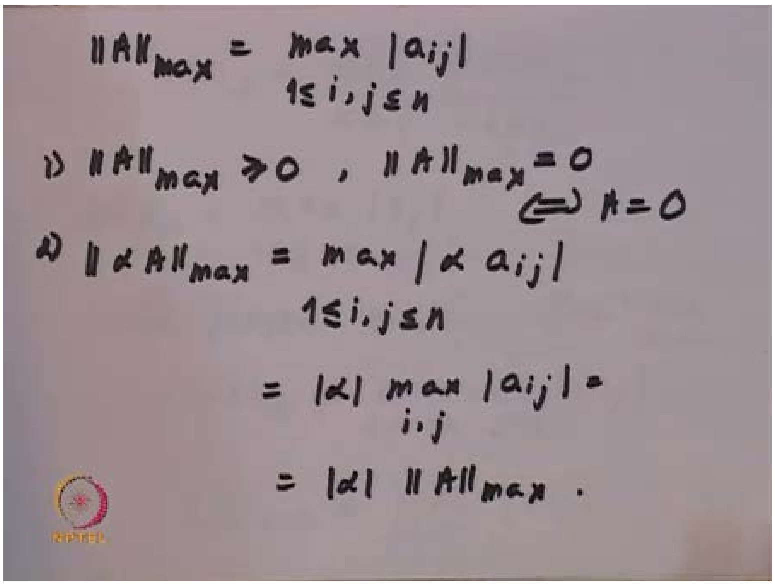
$$\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^{n} |a_{ij}|$$

$$\|A\|_{max} = \max_{\substack{1 \leq i \leq n \\ 1 \leq j \leq n}} |a_{ij}| .$$

So, what was our norm a infinity, it was norm a infinity is maximum of norm A x infinity divided by norm x infinity, x not equal to 0 vector, and norm x infinity is maximum of modulus of x j one less than or equal to j less than or equal to n; and we proved that, norm a infinity is maximum summation j goes from one to n modulus of a i j one less than or equal to i less than or equal to n; so, this is the row sum norm.

And our norm a maximum is equal to maximum of modulus of a i j one less than or equal to i less than or equal to n one less than or equal to j less than or equal to n. So, let us first verify that, it satisfies the three properties of norm and not the forth property.

(Refer Slide Time: 42:15)



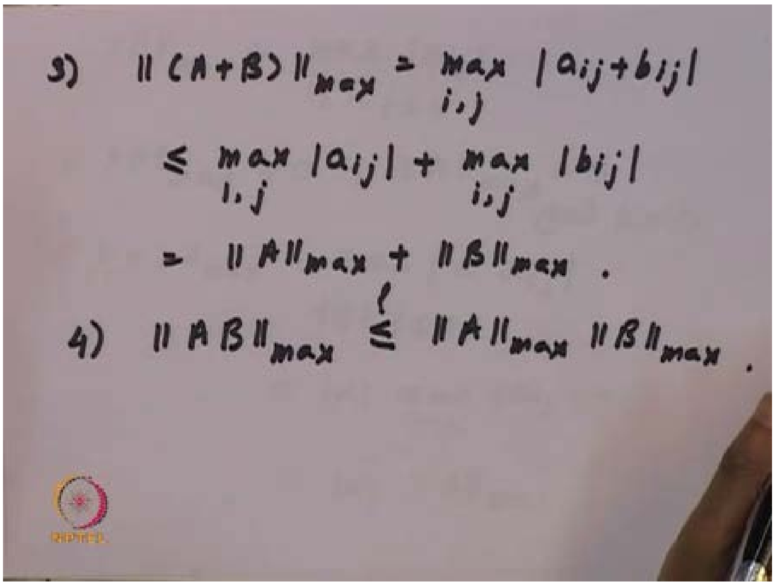
$$\|A\|_{max} = \max_{1\le i,j\le n} |a_{ij}|$$

1) $\|A\|_{max} \geqslant 0$ , $\|A\|_{max} = 0 \iff A = 0$

2) $$\|\alpha A\|_{max} = \max_{1\le i,j\le n} |\alpha\, a_{ij}|$$
$$= |\alpha| \max_{i,j} |a_{ij}| =$$
$$= |\alpha|\ \|A\|_{max} .$$

So, we have norm a max which is maximum of modulus of a i j 1 less than or equal to i j less than or equal to n. So, norm A max will be bigger than or equal to 0 and norm A max will be equal to 0 if and only if A is A 0 matrix; then second is norm of alpha A maximum; if you multiply matrix a by number alpha this will be maximum of mod of alpha a i j, because the each entry a i j will get multiplied by alpha 1 less than or equal to i j less than or equal to n, which will be equal to mod alpha times maximum of modulus of a i j maximum over i and j, so this is equal to mod alpha times norm A max. So, this second property satisfied.

(Refer Slide Time: 43:34)



3) $$\|(A+B)\|_{max} = \max_{i,j} |a_{ij}+b_{ij}|$$
$$\le \max_{i,j} |a_{ij}| + \max_{i,j} |b_{ij}|$$
$$= \|A\|_{max} + \|B\|_{max} .$$

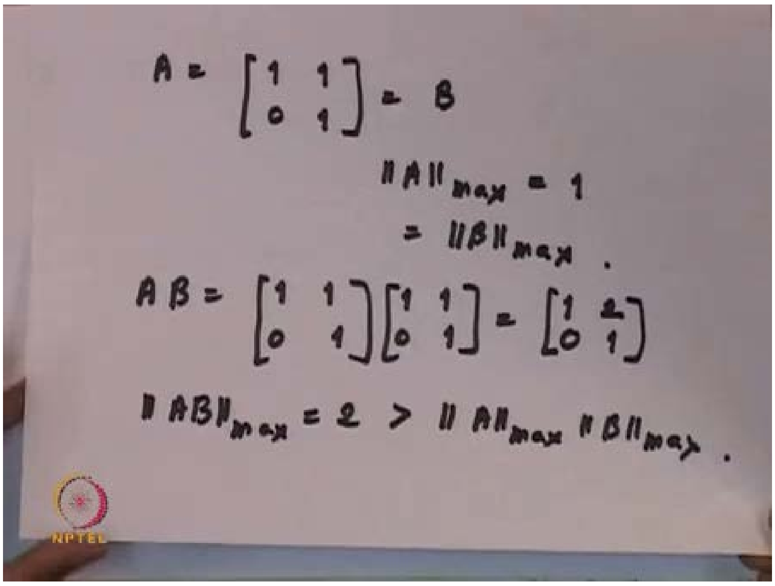
4) $\|AB\|_{max} \le \|A\|_{max}\, \|B\|_{max}$ .

Now, the third property is the triangle inequality; so, we have norm of a plus b maximum this will be equal to maximum of modulus of a i j plus b i j, maximum over i and j; this will be less than or equal to maximum over i j mod a i j plus maximum of mod b i j over i j, so this will be norm A max plus norm B max.

Now, the question is, whether norm AB max, whether it will be less than or equal to norm A max norm B max; now, this is not true and when one constructs counter examples one should always try for matrix of a small size in order to reduce your work or simplify your work, so try for 2 by 2 matrix and see whether it work.

So, what we want is, we want a 2 by 2 matrix or we want two 2 by 2 matrices, such that, norm of a b is going to be strictly bigger than norm a into norm b; now, what is our norm a, it is maximum of all the entries. So, let me look at say a matrix 2 by 2 matrix 1 1 0 1, then its maximum norm a max is going to be equal to 1; now, take b to be equal to same matrix; when I am going to multiply these two matrices, then in the multiplication I will get one of the entry as 2 and that will make norm AB max to be strictly bigger than norm A max into norm B max.

(Refer Slide Time: 46:14)



$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = B$$

$$\|A\|_{max} = 1 = \|B\|_{max} .$$

$$AB = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}$$

$$\|AB\|_{max} = 2 > \|A\|_{max} \|B\|_{max} .$$

So, here is our A which is 1 1 0 1 and that is also equal to B. So, norm a max is going to be equal to 1, which is equal to norm B max; and let me look at AB, so it is 1 1 0 1 multiplied by 1 1 0 1, so this will be the first entry will be 1 first row into first column, second entry will be 2 first row into second column, then we will have 0 here, and then we will have 1 here. So, norm AB max it is going to be equal to 2, because it is biggest entry and this is strictly bigger than norm A max into norm B max, this condition does not hold for the max norm which is the analog of infinity norm for the vector.

So, now, in the next lecture we are going to start a new topic and that is solution of non-linear equations. Thank you.

.