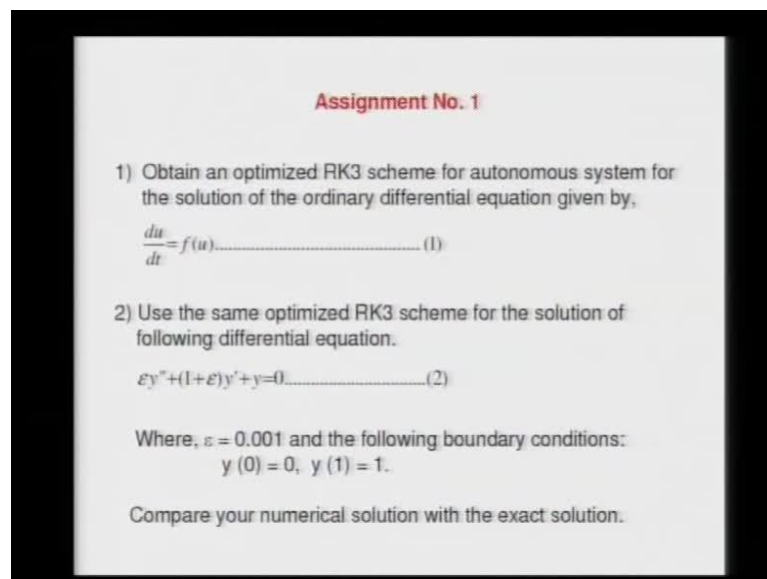


Foundation of Scientific Computing
Prof. T. K. Sengupta
Department of Aerospace Engineering
Indian Institute of Technology, Kanpur

Lecture No. # 22

(Refer Slide Time: 00:17)



Assignment No. 1

1) Obtain an optimized RK3 scheme for autonomous system for the solution of the ordinary differential equation given by,

$$\frac{du}{dt} = f(u), \dots \dots \dots (1)$$

2) Use the same optimized RK3 scheme for the solution of following differential equation.

$$\varepsilon y'' + (1 + \varepsilon)y' + y = 0, \dots \dots \dots (2)$$

Where, $\varepsilon = 0.001$ and the following boundary conditions:
 $y(0) = 0, \quad y(1) = 1.$

Compare your numerical solution with the exact solution.

This is our twenty-second meeting, and today we are assembled here to discuss about the first assignment, which actually has two parts. In the first part, you were asked to obtain an optimized three stage Runge Kutta scheme for an autonomous system for the solution of the ordinary differential equation as given by equation 1. Please note that autonomous equation implies, that the right hand side of one is not an explicit function of time, that is, a definition of autonomous system.

Now, in the second part, we wanted to use the optimized RK 3 scheme, that you have obtained in the first part, to solve the following differential equation. This differential equation, as you notice, has a small parameter epsilon, whose value is given as 0.001 and this small parameter multiplies the highest derivative y'' in this equation.

So, you can anticipate that, this is going to be one of the traditional boundary layer equations, and you have to solve this subject to the boundary condition given below, that is y at t equal to 0 is 0, and y at t equal to 1 is equal to 1.

(Refer Slide Time: 02:01)

The exact solution of Equation (2) is:

$$y(x) = \frac{e^{-x} - e^{-1/2}}{e^{-1} - e^{-1/2}}$$

Runge Kutta method for solving the autonomous equation:

$$\frac{du}{dt} = f(u), \dots (1)$$

We use the following three slopes for the ODE as:

$$k_1 = h f(u^n)$$

$$k_2 = h f(u^n + a_{21}k_1)$$

$$k_3 = h f(u^n + a_{31}k_1 + a_{32}k_2)$$

The RK3 method is defined as:

$$u^{n+1} = u^n + w_1k_1 + w_2k_2 + w_3k_3 \dots (A)$$

So, having obtained this numerical solution, you should be able to compare it with the exact solution. We have already indicated, that the exact solution of equation 2 is given as the quotient of these two functions, which are essentially the exponential functions involving, which you have indicated in terms of x , but please do understand this x and t are synonymous here.

And the Runge Kutta method that you are trying to use here, given by equation 1 involves calculating 3 slopes, and the slope at the beginning point u_n given by k_1 the first relation, then you would get a second slope, which is evaluated at u_n plus a fraction of k_1 , and finally the third slope k_3 is obtained somewhere, which is different from u_n by the two quantities, given in terms of $a_{21}k_1$ and $a_{31}k_1 + a_{32}k_2$.

Having obtained these 3 slopes k_1 , k_2 , and k_3 , we defined the general Runge Kutta three stage method, as given by equation a, that is u at the new time level is equal to u at the old time level plus the weighted average of all these three calculated slopes k_1 , k_2 , and k_3 .

(Refer Slide Time: 03:36)

Hence the parameters of RK3 method are: a_{21}, a_{31}, a_{32} and w_1, w_2, w_3 .

$$k_2 = hf + h^2 a_{21} f'_u + \frac{h^3}{2!} a_{21}^2 f''_{uu} + \frac{h^4}{3!} a_{21}^3 f'''_{uuu}$$

$$k_3 = hf + h^2 (a_{31} + a_{32}) f'_u + h^3 \left[a_{31} a_{21} f''_{uu} + (a_{31} + a_{32})^2 f''_{uu} \frac{f''_{uu}}{2} \right] + o(h^4)$$

Equate (A) with Taylor series expansion for different orders of terms:

$$u(t^{n+1}) = u(t^n) + hf + \frac{h^2}{2} f''_u + \frac{h^3}{3!} (f''^2 f_{uu} + f''^2_{uu}) + o(h^4)$$

$$o(h): \quad w_1 + w_2 + w_3 = 1 \dots\dots\dots (i)$$

$$o(h^2): \quad w_2 a_{21} + w_3 (a_{31} + a_{32}) = 1/2 \dots\dots\dots (ii)$$

$O(h^3)$ terms are with f''^2_{uu} and f''^2_{uu} and they provide,

$$f''^2_{uu}: \quad w_3 a_{32} a_{21} = 1/6 \dots\dots\dots (iii)$$

$$f''^2_{uu}: \quad w_2 a_{21}^2 + w_3 (a_{31} + a_{32})^2 = 1/6 \dots\dots\dots (iv)$$

So, the usual way in defining this problem, as we have discussed in the trial lectures is to basically obtain this parameters of this R K 3 method, which involves finding out those three a_{ij} (s); in this case, they are a_{21} , a_{31} and a_{32} plus those three weights, that we have indicated as w_1 , w_2 and w_3 .

Having obtained the expression for k_2 and k_3 , we can actually expand it in terms of a series, as given here as k_2 is equal to $h f$ plus $h^2 a_{21} f'_u$, then h^3 square by factorial 2 $a_{21}^2 f''_{uu}$, and h^4 by 3 factorial $a_{21}^3 f'''_{uu}$ times, f cube times the third partial derivative of f with respect to u . Please, understand that since we are interested in a third order method, we irritate term after h^4 to get the truncation error term, which is going to be proportional to h^4 .

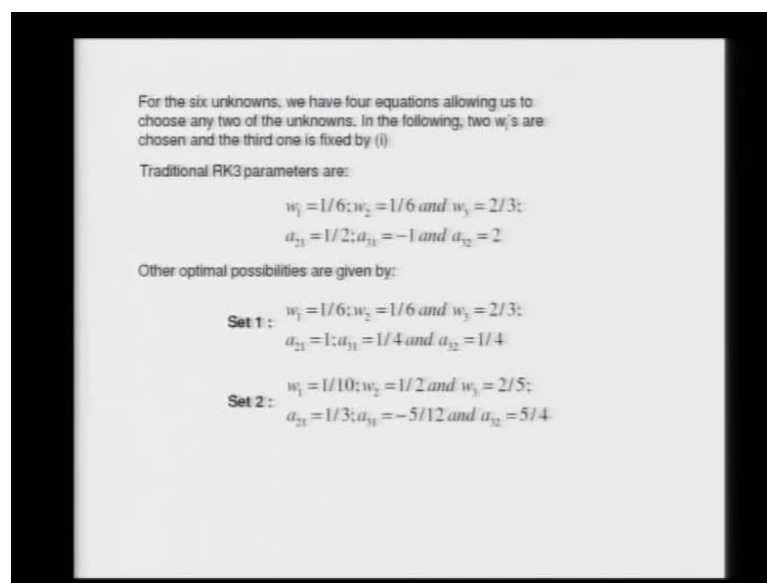
The same way you can expand k_3 in terms of the function f at the starting gate plus these various functions, and its partial derivatives with respect to u , at different points which involves all this unknown coefficients a_{31} , a_{32} , etcetera. Again this we indicate all the way up to h^q term. Now, what we have noted, that we could also write down the Taylor series, at the new time level t_{n+1} , in terms of $u(t_n)$ plus this other terms indicated here.

If we equate this Taylor series here with the term given here as a , then we would be able to equate the coefficients of various terms involving a powers of h , for example, if we

equate the coefficients of terms proportional to h , then the Runge Kutta method will give you w_1 plus w_2 plus w_3 while the Taylor series will just indicate, that is just function is coefficient is 1.

The same way for order h square term, we get the equation 2 here, and for the order h cube term will have two sets of terms, one will be multiplied by $f f u$ square, and another would be multiplying f square times $f u u$, and when you equate those coefficients you get this two relations, which I have given by 3 and 4.

(Refer Slide Time: 06:57)



For the six unknowns, we have four equations allowing us to choose any two of the unknowns. In the following, two w_i 's are chosen and the third one is fixed by (i)

Traditional RK3 parameters are:

$$w_1 = 1/6; w_2 = 1/6 \text{ and } w_3 = 2/3;$$

$$a_{21} = 1/2; a_{31} = -1 \text{ and } a_{32} = 2$$

Other optimal possibilities are given by:

Set 1:

$$w_1 = 1/6; w_2 = 1/6 \text{ and } w_3 = 2/3;$$

$$a_{21} = 1; a_{31} = 1/4 \text{ and } a_{32} = 1/4$$

Set 2:

$$w_1 = 1/10; w_2 = 1/2 \text{ and } w_3 = 2/5;$$

$$a_{21} = 1/3; a_{31} = -5/12 \text{ and } a_{32} = 5/4$$

So, basically, what we note that, we have 6 unknowns, and we have generated 4 equations, that allows us choosing any two of them, those are your degrees of freedom, and in the following, I have just indicating one way of doing it, that you choose two of the w_i (s), once you do that, then from equation 1, the third w_i is obtained, and you can solve for those a_{ij} (s), and in the traditional R K3 method, this parameters are given by a_{21} is half a a_{31} is minus 1 and a_{32} is equal to 2. What we have done gone through this exercise, we have performed an optimization and looking at the next higher order truncation term, and that we try to solve, and we obtained the following two sets set 1 and set 2.

So, essentially, this is what our solution methodology is going to be defined by, so we can choose any one of the methods, and obtain the solution, and **d that is what, r**est of you,

some of you are going to make a presentation, and let us all sit back, and listen to their talk.

Adithya, would you please come.

(Refer Slide Time: 08:45)

Optimized Third order accurate RK:
TWO possible solutions

	Optimized Set 1	Optimized Set 2	Standard RK3
w_1	$1/6$	$1/10$	$1/6$
w_2	$1/6$	$1/2$	$4/6$
w_3	$4/6$	$2/5$	$1/6$
a_{21}	1	$1/3$	$1/2$
a_{31}	$1/4$	$-5/12$	-1
a_{32}	$1/4$	$5/4$	2

Which of the optimized solutions is more accurate?

Good morning everyone, I will like to post to summary of my key observations, which I do not know the solution of assignment 1. So, first see when you solve a set of equation, you take 4 equations from equating the orders of h , h^2 and h^3 , and you have 6 elements which are on the left panel w_1 , w_3 , and a_{21} , a_{31} and a_{32} , and therefore, the system is not closed, and to uniquely solve for all these unknowns, we required two more equations.

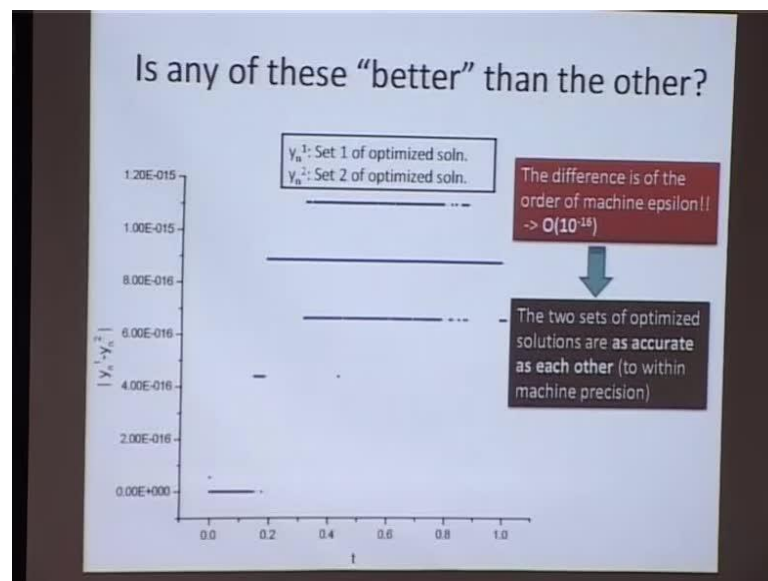
So, these three equations are obtained from organizing the truncation error. What I mean by optimizing the truncation error is writing truncation error in terms of three parts, where in one of them is a constant, it does not have any variables w_1 to w_3 , a_{21} to a_{32} , and the other two terms have are functions are some functions of these 6 variables. So, basically what we do is, we equate those two terms to 0, and therefore, system is not flows and you can uniquely solve for these.

It turns out it, when you do that, you end up with two sides of solutions, which are your optimized solutions, which are shown in the first and the second column, it is quite surprising, that values of these constants is just so far from each other, **in**, I mean, you

can, if you look at that first and second panel, and compare to the standard R K 3, which I got from hopefully some reliable sources, it is quite clear, that the weight different with respect to session, that if you look at a 2 1, it is equal to 1 in the set 1, and equal to 1 by 3 in set 2, which means that, if you look at set 1, what it does is, it shoots across the entire step size and goes to the next point, and then calculates the slope, whereas in the second method, what it does is, it goes to one third, and then goes to minus 5 by 12, which means, it goes in opposite direction, and then goes plus 5 by 4 h, which means, it shoots to more than h to come back to the third point.

So, basically both these sides of solutions are very different from each other, and that characteristics, but so we look at this is the question was asked here is, which are the two optimized solutions are more accurate, **if at all**.

(Refer Slide Time: 10:50)



So, what are done here is, I have plotted the absolute value of the difference of y_n^1 and y_n^2 , where y_n^1 is the numerical solution obtained from the first set of solution, that is w_1 equals one sixth and so on. And y_n^2 is the numerical solution obtained from the second set of constants, which is 1 by 10, 1 by 2, 2 by 5, so on.

And I plotted the function of p , p which is the independent variable, and this plot, which I have done is further step size of 10 power minus 4, which means, I have taken 10000

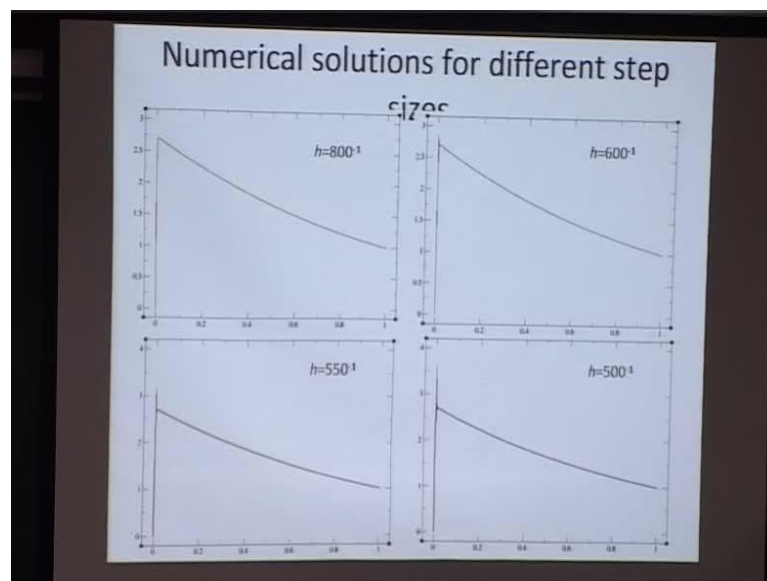
steps, and I have seen, **what the absolute**, how the absolute value of the error between the two optimized solutions.

The two numerically optimized solutions varies with the independent variable. And if you look at the order in the y-axis, it is of the order of 10 power minus 16, **it is**, it does not have a unique value, it is, it ranges between 6 into power minus 16 to around 11 into 10 power minus 16.

Now, it might seem that one method is therefore more accurate in the other, but keep in mind, that we are doing a numerical computation, and therefore, this 10 power minus 16 is basically the machine epsilon, it is nothing but the machine error.

The machine cannot do a calculation, cannot represent something more precisely as 10 power minus 16, and therefore, I believe that these two solutions are as accurate as each other to within machine precision.

(Refer Slide Time: 12:18)



So, it is clear, so what I have done here is I have plotted, so now let us know that both the numerical solutions are as accurate as each other to within machine precision. I just take one of the two sets, I think, this one I have done for a 2 1 equals 1 by 3, that particular set and see how that compares with the exact solution, and done that for the range of steps sizes.

We expect that as you increase the step size, at one particular step size, the solutions stops converging to the exact solution, and shows large deviations from it, and thus exactly what we see. The first 4 plots are as you can see, the first one is for h equals 800 inwards, 600 inwards, 550 inwards, and 500 inwards, you can easily compare the numerical and exact solutions.

If you look at 800 inwards, the numerical solutions, the numerical solution and the exact solution indistinguishable from each other, whereas at 600 inwards, you can easily see that, there is a red spike that is, sort of deviating from the black curve; the red one is the numerical solution and the black one is the exact solution.

(Refer Slide Time: 13:29)

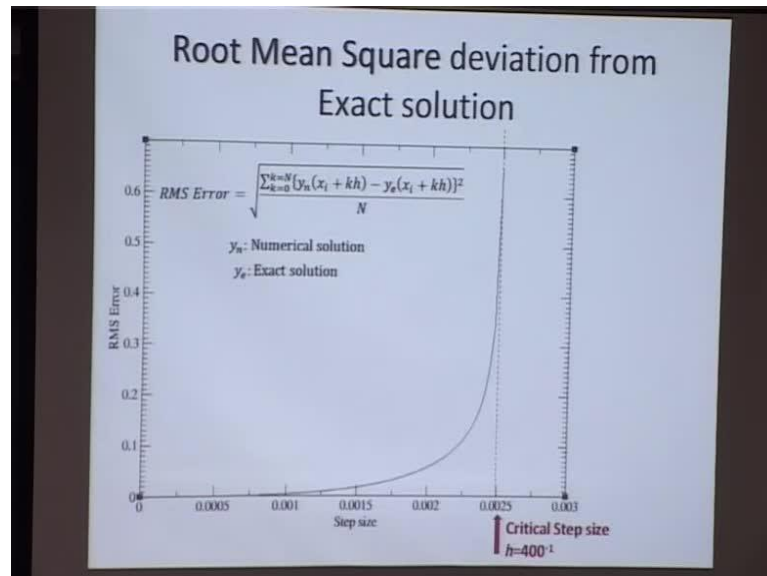


And as we increase h more and more you can see, that the error between the numerical solution and the exact solution keeps on increasing, which is what we expect. These four plots are essentially the key plots for 475 inwards, 450 inwards, 425 inwards, and 400 inwards, which shows very clearly, that the deviation of a numerical solution from the exact solution.

It really increases, very rapidly as increases step size, and the solutions, the programs stops to terminate above a value of h equals 400 inwards or around 395 inwards, which means, that is the critical step size above which you cannot, that is the maximum step

size that you can take for, to have a stable solution, all though not very accurate even at 400 inwards.

(Refer Slide Time: 14:18)



So, it may not be very clear what the exact value of the so-called critical step sizes from these plots, because that taken at few values of h , so what I have done here is, I have plotted the root means square deviation of the numerical solution from the exact solution.

How I do this, I have shown it in the top left insert in the plot, I define the RMS error as that which there y_n at x_i plus kh minus y_e at x_i plus kh , that difference the whole square, and I sum it that over all the steps, and divided by number of steps and take the square root, that exactly how we define the RMS error. And the reason, I take the RMS error is not just the difference of y_n and y_e , is because of this, as you can see, for example, in h equals 400 inwards, the difference between the numerical and exact solution is positive, and then becomes negative, positive, negative, etcetera.

So, it makes more sense to define the error or the deviation of numerical solutions from the exact solution in this particular way. So, this is what I have done, and there y_n is the numerical solution, again like I said, I have just taken one of the two numerical solutions, because I have shown them to be as accurate as each other, to within machine precision.

So, y_n is a numerical solution and y_e is the exact solution, which I have delayed on elliptically x_i is 0, and our $R_K(s)$, it is the initial point of x interval; now, it actually t_i ,

because I have taken, define t as the independent variable value here. I am sorry about that, and k as the index of summation at h is the step size, which is one over n in this case, because it is x_f minus x_i divided by n .

So, basically that is my RMS error, and I plot that as a function of step size, and as we expect the RMS error or the deviation of numerical from exact solution increases.

It is very, very small over the values of h around up till about 10^{-3} , which is 0.001 on the x -axis, but beyond that, it starts to rapidly increase, and you can see that for h equals 400 inwards or 0.0025 , and offered values of h more than that, the RMS error is really high, which means, that is can be sort of defined as a critical step size.

Of course, critical does not mean that, all step sizes up to that will give you sensible numerical solutions. As you have seen h equal to 400 inwards, gives us pretty trashy numerical data; so, it would be advisable to stick to the range of the tiniest RMS error on that graph.

I am just trying to show, I mean, this make sense, because if you look at the nature of the exact solution above particular step size, I mean, there is a very sharp variation of the exact solutions at the peak, right around x , I mean, less than 0.01 , there is a peak in the exact solution. So, whenever the exact solution varies very rapidly, there is a good chance that the numerical one deviates from the exact one; as you can very clearly see, this starts to deviate exactly at the peak.

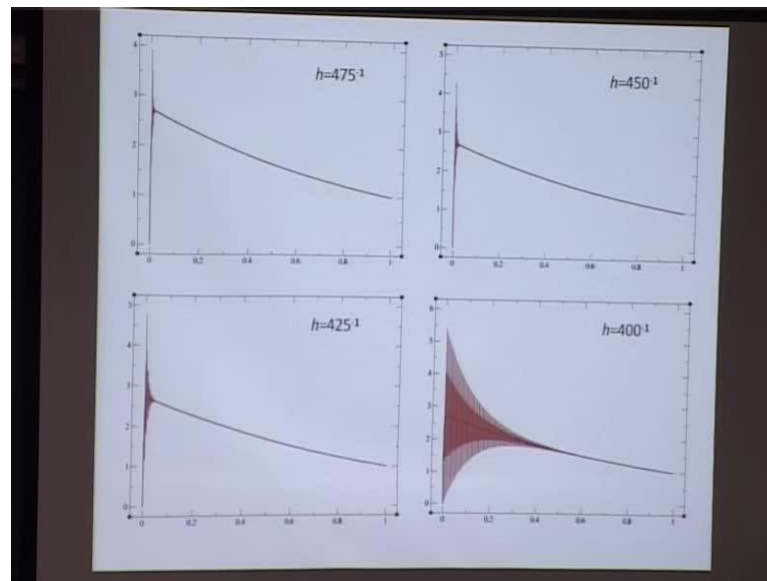
So, for such functions, for such exact solutions, which show very sharp corner or very sharp peak in there graphs; we would really expect that above a particular step size, the numerical round does not converts to the exact solution at all, which is exactly, what we see here in terms of the RMS error becoming very, very high above h equals 400 inwards. So, that is basically summary of my results anybody would.

I think we are now open for discussion, let us open it to the floor, anyone having any observations to.

That is in fluid, that means, $(())$.

Oh no, so.

(Refer Slide Time: 18:29)



For 400 inwards this is the exact solution, mean all other cases, this is the exact solution, we should have indicated with an arrow or something; so, this is the exact solution, this is a numerical solution, so I do not know, why you are having two bands, if that is your question, no.

probably you have three points, at define this peak; you have two of them here, and one on top, so it sort of looks like triangular thing, I mean, that is an artifact of I do not think this anything to do, I mean, I do not think that two bands exactly have any meaning.

It does.

How do they...

How do they mean?

It does right.

Utkarsh.

What does it imply?

Is that first triangle try to (())?

Well, I will interpret what he is trying to say he takes a 50 and we will see that two dominant frequencies, one corresponds to rapid variation, that corresponds to a denser lines and one is a little.

So, also very easy thing to do, for one to do is basically take a 50, you have seen, it is a oscillatory solution, that is have been that of any instability with all, we have starts up with some kind of instability and then it goes beyond control. One thing while explaining about RMS error, you made the observation, but the excursion could be plus or minus, but as you can see, in your case, it was always positive, so it was intended.

So, n minus y can either be plus or minus.

So, you could have taken about, I mean, just absolute value, so that is what is called as n infinity; what you have done is we have shown $|t|$ naught, any other questions?

I was just to find what could be the value of h , mean, how we can get the value of h ?

Well, we have an estimate for your error, you can take a look at it, but again you did not show the expression of truncation error, but if you would have looked at the expression for the truncation error, **you want to**, if you want to take a look at your, you could use your words, and if you could write it down, we would note that the truncation error has two component, one depends on the parameter of the method, however part which is independent of it.

(Refer Slide Time: 21:21)

$$\begin{aligned}
 T^{n+1} &= u^{n+1} - u(t^{n+1}) \\
 &= h^4 \left(f_{nnn}^3 \left[\frac{w_2 a_{21}^3}{6} + \frac{w_3 (a_{31} + a_{32})^3}{6} - \frac{1}{24} \right] \right. \\
 &\quad \left. + f_{nn}^2 f_{nn} \left[\frac{w_3 a_{21}^2 a_{32}}{2} + w_3 a_{21} a_{32} (a_{31} + a_{32}) - \frac{1}{6} \right] \right. \\
 &\quad \left. + f_{nn}^3 \left[-\frac{1}{24} \right] \right)
 \end{aligned}$$

So, in that sense, this is not really absolute optimal, because (()).

No, that is all you hold on let him finish writing, and see you get where the fifth and sixth from this truncation error expression only that is h^4 , and you can see that, he is writing the leading term is order h^4 .

Well, he writes rest of you can continue the discussion.

The number of truncation terms u t (()).

So, it is all, was like that, that is the way you define truncation error, is the next neglected set of terms.

So, what he is done here, you can already see there are 2 sub sets of terms, which depend on this coefficients right w_1, w_2, w_3 and a_{21}, a_{31}, a_{32} .

So, those are the stool sets, so what I suppose most of you have done is could this equal to 0, and that is one way of doing, is there any other way you could have done it?

Yes, shakthi.

(())

Such, it there differentiation of what.

First.

This error with respect to w_2 and w_3 .

So, that is another way of doing it, is you convert those 4 equation into two parameter family of solutions, so arbitrarily you have to choose.

Well, it is probably, if you plot the truncation error term with the variation of this coefficient, it probably have a global minimal there with respect to those two terms, but there could be a termed alternative, I would leave that towards the end, so let other people can also get in, and so, we will thank Adithya.

He has also going to the second set of those values for w_1 , w_2 .

Already, he had $(())$.

Why you check the, actually it turns out to be a quadratic invoke some of the parameters.

$(())$ quadratic, and a 2^1 , and therefore two values a 2^1 .

In fact, I think u naught three, Sonam.

One was inconsistent.

Well, how do you say, it is inconsistent.

So, why should that really doubt, if w_t is 0 that is fine, I would require less computation.

So, w_3 2 two variables is 1.

So, one of the condition would be violated that is fine, thank you Adithya. We should now move on to the next one, now its turn Himanshu.

You have the microphone and the chock, you can probably it is good to keep this expression in front, I mean, the bottom left there is a cup, click on that, the third, no, that is it.

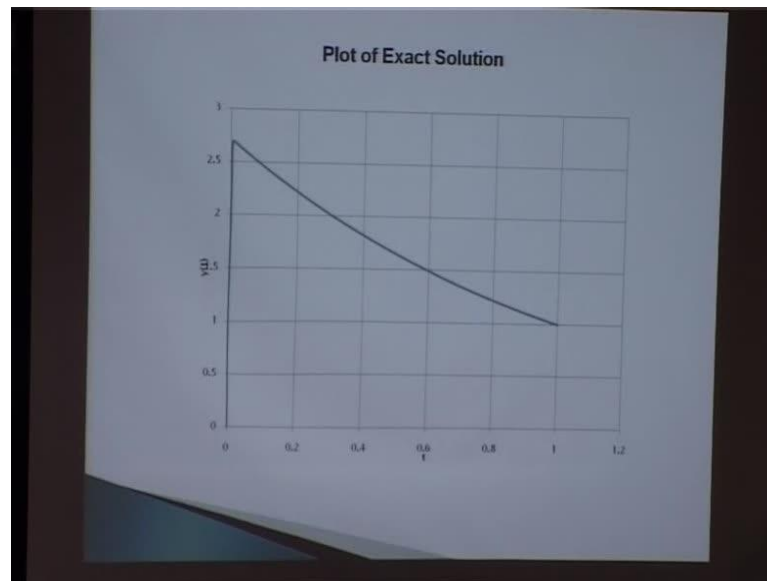
(Refer Slide Time: 26:54)

Two Numerical Solution Sets	
Solution Set 1	Solution Set 2
‣ $W_1 = 1/10$	‣ $W_1 = 1/6$
‣ $W_2 = 1/2$	‣ $W_2 = 1/6$
‣ $W_3 = 2/5$	‣ $W_3 = 2/3$
‣ $a_{21} = 1/3$	‣ $a_{21} = 1$
‣ $a_{31} = -5/12$	‣ $a_{31} = 1/4$
‣ $a_{32} = 5/4$	‣ $a_{32} = 1/4$

Good morning everyone, well the problem that we had was, we had to second order differential equation with boundary values, and we had to find, we had actually to optimize the R K 3 method for the computation of, for the numerical solution at various values of t . Now, again while solving the equations, we get three set of solutions, in one of which a a_{21} comes out to be 2 by 3.

When you put a a_{21} equal to 2 by 3, then but other equations, you find that system becomes inconsistent; so, that value can be neglected. The other two sets have been shown here, where a a_{21} is the other, that has been elected first, a a_{21} is equal to 2 by 3. In these sets, in one of them a a_{21} is equal to 1 by 3, and in other one a a_{21} is equal to 1, and these are the solution at Adithya have point it out.

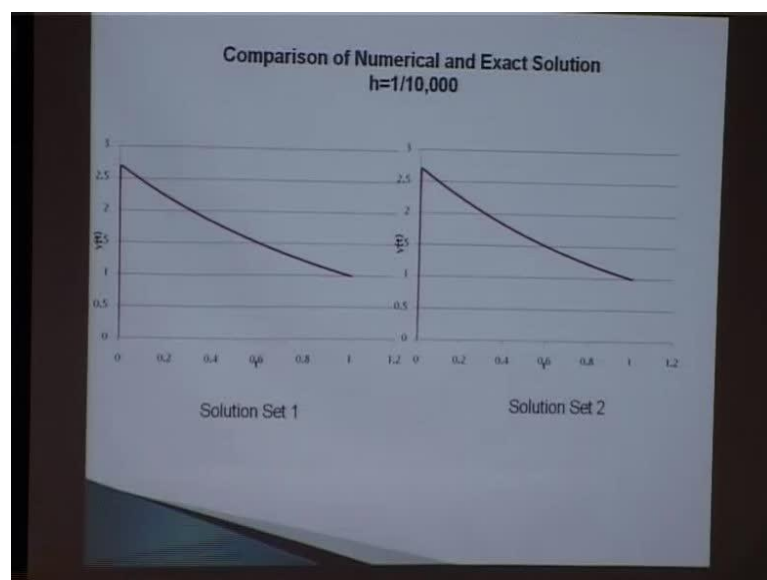
(Refer Slide Time: 26:56)



Next, if you plot the exact solution, it looks like this value, so we can clearly see, it is actually a boundary value problem, the gradient here is very sharp, it shoots up to a value of 2.6 in a very small interval, and then it decays very gradually.

So, it is actually boundary value problem, and so we can see that the equation is a stiff equation, and the value the step size would determine the nature of the solution, that we are actually getting.

(Refer Slide Time: 27:27)

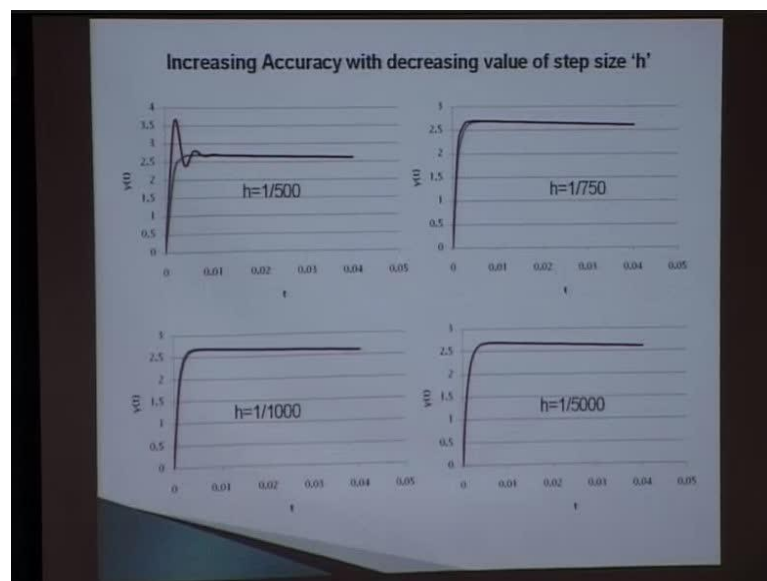


Like we said, we get the 2 solution sets, so while comparing to 2 solution sets, this is the plot of the numerical error, the red one shows the numerical error, and the numerical solution, and blue is the exact solution, that we have plotted using the code in c.

So, this is the kind of plotted that we get, this is the exact solution, this is the numerical solution, so we can see some kind of wiggling here as Adithya also pointed over it, h equal to very large value, it is 500 inwards to this same kind of wiggling here. We can see similar kind of wiggling in the second solution set.

As we decrease h value, it is 1 by 1000, so the wiggling decreases, but still the solution is not exactly correct; because this is the region of greater solution differs. As we still increase h the solution, all most overlaps with exact solution, and it happens in both the solution set.

(Refer Slide Time: 28:31)



So, we can see that the both solutions at almost identical. Now, since both the solution sets are almost identical, so we take one of them, and we show the magnified view.

Now, this (()), so this was a region of consideration, where we were showing some kind of wiggling; so, now we will magnify that thing.

So, and this, the value of h is 500 inwards, and blue is the exact solution, the red one is the numerical solution; so, this kind of oscillation we get. As we decrease the h value, there is some difference still present in this interval, we again decrease it; again, we can see some very fine difference between the two sets.

(Refer Slide Time: 29:59)

1/h	Solution Set 1	Solution Set 2
500	0.02473520286859477	0.02473520286859501
750	0.005665489791910013	0.00566548979191019
1000	0.0020029336892512127	0.0020029336892510535
2500	4.1692165506317097E-4	4.162165506519355E-4
5000	2.0092758011302302E-4	2.009275801160935E-4
7500	1.3355325756142308E-4	1.3355325756435958E-4
10000	1.0008568926802323E-4	1.000856892793039E-4
50000	1.0038835886591687E-8	1.0038835984044576E-8

If the blue is above the red one, but h increase, h decreases further, these two solutions almost overlap. So, we can see that there is some value of h , optimum value of h , below which the solution would be this good and above which like, somewhere around 10 raised to minus 3 , if we do not, if the value of especially take is the greater than 10 raised to minus 3 , then the solution that we get is not exactly a correct one; these were the relative RMS error, now how we calculate a relative RMS error.

Is it any different than what Adithya did?

(())

At equal to zero, we do not take, because that is a point.

Well, that is anyhow 0 , any way it does not contribute.

(Refer Slide Time: 29:59)

1/h	Solution Set 1	Solution Set 2
500	0.02473520286859477	0.02473520286859501
750	0.005665489791910013	0.00566548979191019
1000	0.0020029336892512127	0.0020029336892510535
2500	4.1692165506317097E-4	4.162165506519355E-4
5000	2.0092758011302302E-4	2.009275801160935E-4
7500	1.3355325756142308E-4	1.3355325756435958E-4
10000	1.0008568926802323E-4	1.000856892793839E-4
50000	1.0038835886591687E-8	1.0038835984044576E-8

So this is the relative errors that we get from solution set 1 and solution set 2. Again, it depends up on the accuracy of the machine that we using so far, h equal to 500 inwards this was the error of set 1 and set 2, we can see that these figures are same, what differs is these values; similarly, for the various values of h , so these figures are same.

It is do not (()).

We only differing in the last values, and when we actually try to look, what is the order of this error, the difference in this, in that, that actually comes out around certain points 14, for some instance point 15, for some...

So, did you do a double precision calculation or single?

Double precision calculation, and this shows that, it is not only the machine error, there is a little bit of difference, when they do solution. Again, here the solutions are same with this figure, but then this start, and that is also 10 to the power minus 8, so that is what 10 to the power minus 15 precision, the two errors are same up to 10 to the power of minus 15.

Can you make a guess, why it could be?

Anyone, in the class.

Sir, again that depends on only the, I mean, this difference in this coming, we are taking the, I mean minimizing term.

So, up to this order magnitude, which is why the effect comes after the 10 to the power 10^{-8} .

So, I mean, what are you saying.

No, but this is we have done the same similar kind of truncation or minimization for both the sides, we minimize the truncation error, and 10^{-8} , the order of other quantities is truncation error; it comes prominent only after the 15 decimal place.

No, it is happening much before, if you look at the 50000 data, I think, it is in the seventh or eighth decimal, my question is why they are different.

10^{-8} , so that is 10 to the power minus 8, that is, what the 10 to the power minus 8.

Does not matter, that is, see when we talk about rounding of, we talk about significant figure not the exponent, keep the exponent out of the picture; so, that is, where you are stay in a single precision domain 10 to the power minus 8, it is not then domain precision, you should be able to get up to 10 to the power minus 14.

So, my question to you, all of you to think, because Adithya also mentioned machine epsilon, what is machine epsilon, you have done two courses, some of you, but all of you have done the first course on computing.

So, what is machine epsilon, what is this, what animal is this.

10^{-8}

Shall, we have some one else to answer.

Someone 10^{-8} was trying to say something.

10^{-8}

So, what is this error due to machine?

Sir, these are numbers can $(())$.

So, is this some kind of a sort of a value for a machine or $(())$.

I will say that you cannot represent any $((\text{no audio 33:53 to 34:15}))$, in any language, they say double a, b, when we write if a equal to equal to b, it means, if mod of a minus b is less than epsilon, when the machine is $(())$.

So, basically I am asking what throwing that the question at you, are you saying there is a lakshman, rekha behind which we cannot do given a machine.

That does it on the register side, and it does not be, and only has a certain number of digits.

That is not correct, have you heard of lawn double?

Yes, sir.

Quadratic precision, yes sir, you can define, and you can store the numbers, so that happens here is the same number $(())$.

No, please try to understand, what they actually done, when you store a number, the real number, he assigns some storage space for each significant digits. Single precision you would be talking about some of digits let us say for some particular hardware.

Double precision could be let us say 14 digits, lawn double, it could be 18 digit, it could be 21 digit. So, what happens when I am defining a quantity a double precision, what I am doing for each number, actually I am allocating two spaces; so, I could do much more, I could do define quadratic precision, so there is no such seen as machine epsilon.

So, it is something which depends on our ability or our desire to express a number, we can do it, but at the cost of more memory, and more computing time, and the thing that you are talking about having decided up on the significant digit, you want to represent; so, you have basically a rounding the number of beyond those digits.

So, this could be a better term to call it a round off error. So, you can have a different kind of round off error, for different declaration of precision, I am sure, it must have been taught in your first three course in computing.

You may mention that you have done some calculation these are coded in c, did you did java?

Yes, sir, java.

Is there anyone, you may have try to do it in two different languages, and seen if Utkarsh, you did what about the speed, did you see any...

No difference may be this problem is too small.

Sir, series are always considered $(())$, sir we have different results $(())$.

Identical.

Identical in java, she has done it in $(())$, we are getting different results, we are not getting the $(())$.

So, we still have not answer that question, why those two listen tries are different in the seventh or eighth decimal place, when the calculations are done in double precision, why would they be different.

Because as you can see, we have identical prescription up to h 4.

$(())$ both are same steps.

No, not the steps, I mean, sorry we are taking a 21 value.

In one case, we have value equal to 1, whereas in the other, we have 1 by $(())$.

(Refer Slide Time: 37:54)

Two Numerical Solution Sets	
Solution Set 1	Solution Set 2
‣ $W_1 = 1/10$	‣ $W_1 = 1/6$
‣ $W_2 = 1/2$	‣ $W_2 = 1/6$
‣ $W_3 = 2/5$	‣ $W_3 = 2/3$
‣ $a_{21} = 1/3$	‣ $a_{21} = 1$
‣ $a_{31} = -5/12$	‣ $a_{31} = 1/4$
‣ $a_{32} = 5/4$	‣ $a_{32} = 1/4$

No, but at the same time, your error is uniquely defined, in both the cases, those two terms has been see, that is why I am not surprised, that those two sets of results are almost identical.

I am asking you through back question, why it is almost are not exactly; now, answer must be staying at your face, any one.

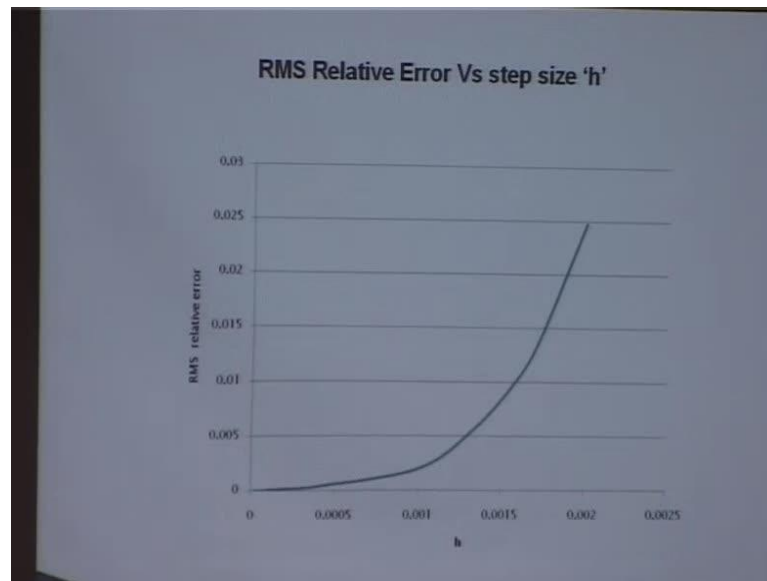
Sir, that is probably, I mean, that is where when the machine is calculating, it is rounding on ((.)).

Are we activating it to round offer, then again, we do not agree.

One more sir, I am having one more reason, since it, we are using the shooting method; so, the error might be 0 for a same value, if it is a same value of initial z primes, then the error should be the same, but when we are doing it my shooting, we are not exactly saying that, they are converging to the same value in the same fashion.

Then, we will go to a next presentation, and then towards the end, I will come back to it.

(Refer Slide Time: 39:03)



I think we will go to a next.

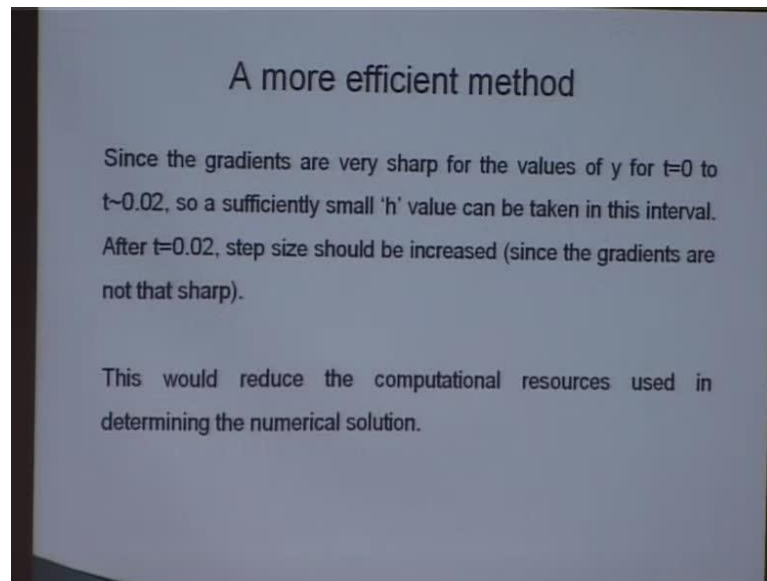
This is the...

This is the brought up the relative RMS with the h as...

You know, what I would suggest to all of you, whenever you are looking at error, like this, please use log scale, otherwise you are not able to distinguish between different cases.

I have taken...

(Refer Slide Time: 39:35)



We feel that more efficient method have, could have been which same like in the region of p equal to 0 to 0.02 to 0.03, that is a region of the consideration, where the gradient very large, but if we take h to be uniform, like very small for the higher values of t also, then actually we are wasting the computation resources. So, an efficient method would have been to take the sufficiently smaller value of p for smaller values of some shift sufficiently small values of h for t small, and as t increases, since the gradient is not that sharp, so we can even increase the h value, so that would have given.

So, we are all lazy, we did not think and did you, all you did.

Is it 20 here, whenever we are doing the transition error minimization, we are going through all this algebra, what I did was, I take by using the traditional coefficients.

No, I do not believe what is traditional.

No, sir let (()).

Are you too modern?

No sir, I am simple, basically my taking any arbitrary coefficients.

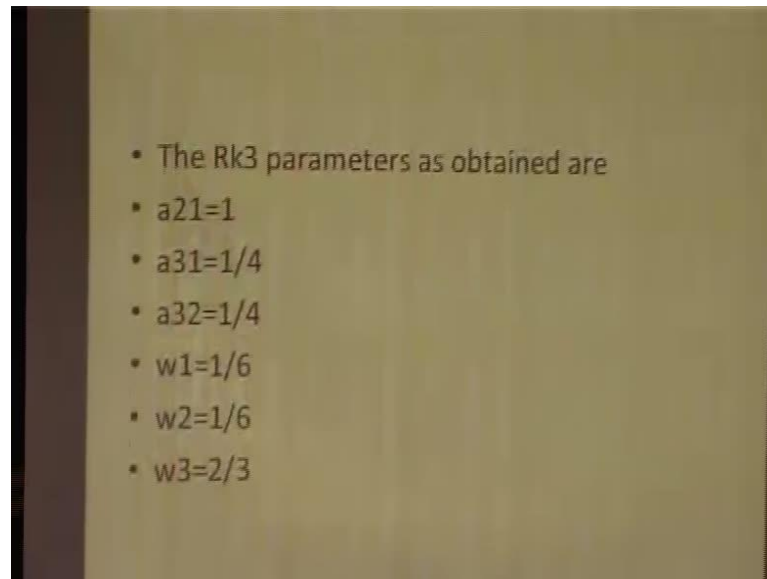
And taking the optimal coefficients, the amount bulkily improves the error, it is much lesser than doing, I mean, decreasing a temperature.

Has anyone obtained solution with standard method and compared the error?

Yes.

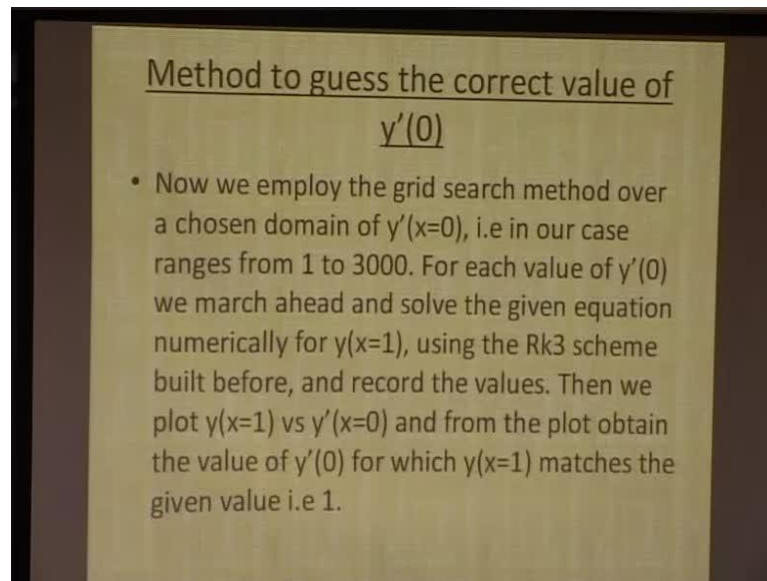
You did, so again will come, why you just come in will go to a next.

(Refer Slide Time: 41:06)



Most of the things have already been discussed like obtaining out how to obtain the exact solution, make the solution all those things, here the parameters which you have, we were worked upon (()).

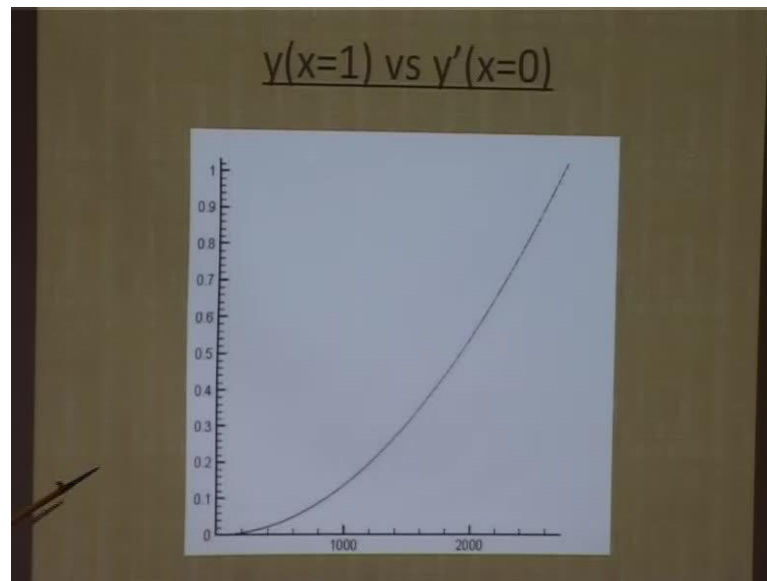
(Refer Slide Time: 41:11)



So, this is how we like to move on, like we had to assume certain value for y prime at x equal to 0, in order to match a head to compute the solution numerically, so let we implied the great search method, in which, we like, we just checked at a particular domain for y prime equal to 0, that is from 1 to 3000, because and this is very chose, because if you like go for the exact solution, you will get y prime 0 to be around 2715 some value like that.

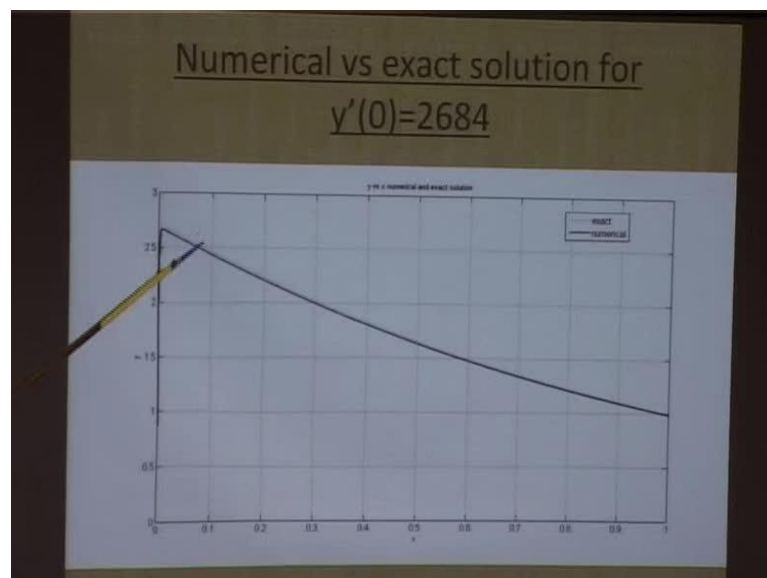
So, that is why we chose our domain to be like this, and for every value y prime we obtain the value of y at x equal to 1, and like that shooting method, you will see if it matches, then it is better, otherwise we have been go to the next value, like we obtain values of y at x equal to 1, for all the values of y prime at x equal to 0, and we plotted it.

(Refer Slide Time: 42:16)



This is the plot for y at x equal to 1 versus y prime at x equal to 0. You can see like for if we want to have the like given the boundary condition at y at x equal to 1, which is equal to 1, so that value, if we like interpolated, we will get some value, numerically we get, I have got the value of 2684 which is quite close to that value.

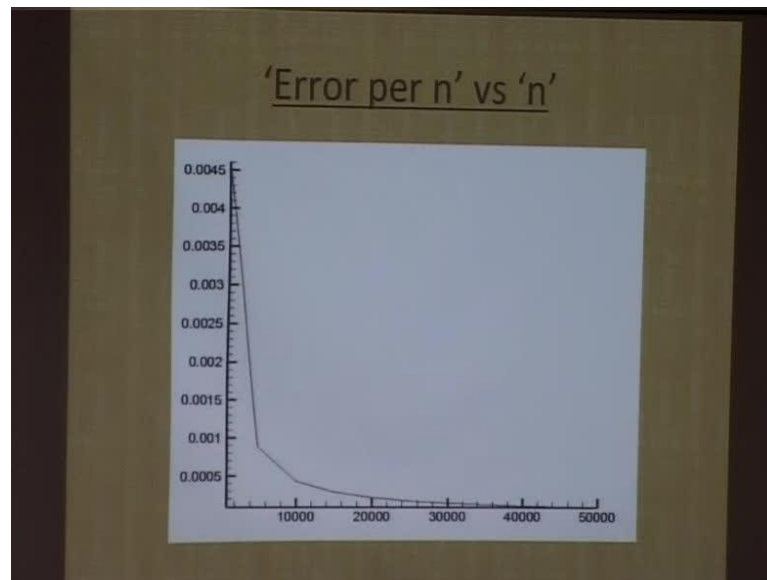
(Refer Slide Time: 42:48)



So, that value this is the plot for the numerical versus exact solution for 2684. This line shows the numerical solution, while this dotted line, if you can see it shows the exact solution.

So, there is some difference between the exact and numerical solution, and if that difference, like what has been done here, that is the root mean square of that error, that the difference between the exact and the numerical solution, if we plotted.

(Refer Slide Time: 43:19)



If we plotted, we will get us something for and this is make error per m versus n, that is n is the number of grids, that we imply n versus n; so, this is how we get now, and one important thing using this graph is like, we can use this graph to in order to determine, what should be our grid number of grid points to be implied or rather what should be a great size.

As you can see that, as we go on increasing the number of grid points, the error per n decreases, and soon it this can give us an stimulate of how many grid points, we can imply in order to get as close to the exact solution of problem.

That is provided, you take h as same.

That is why.

Provided h.

Provided space s a.

Spacing will gone.

So, I mean based on that, we just now heard, that would be a probably good idea to keep it in the inner part of the boundary layer.

Yes sir.

In the outer part, we could do little more liberal.

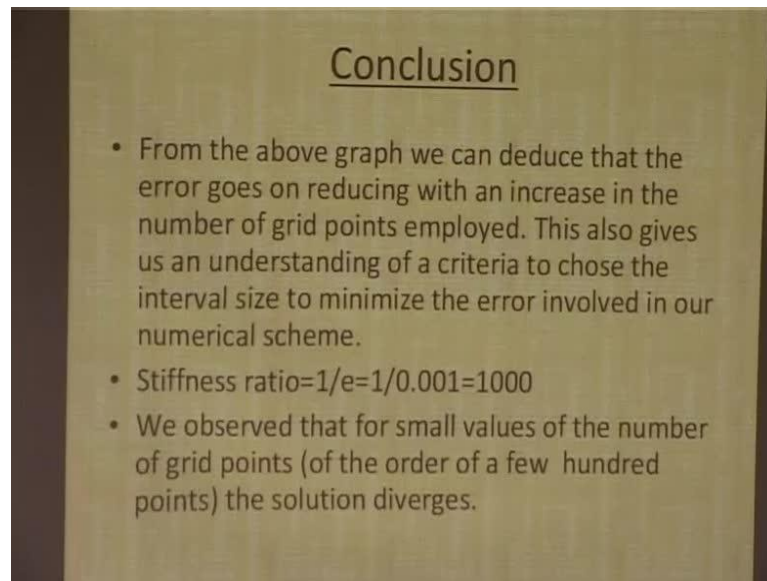
And probably, even that 400 limit, is not a correct limit, if I have to do it, I would get the solution in 200 points by having variable space, so that is how it should be.

So, if like, if we consider, this is just form of solution which you are getting, so this is quite like a boundary value problems, so what we are talking about is, like if we derive like this, it mean this whole graph into two parts, like we can solve for this domain, we can have a different grids system, and for this domain, you can have a different grid system for this; since the changes quite gradual, we can have like the grid spacing could be not that, like refined as for this case.

So, like we can have a non uniform grid system. in order to solve this problem, that could be much faster than, like having a uniformed system, and like refined grid spacing for the entire region.

Coming to the conclusion part, it is clear ,it is quite clear on the graph, that has been keep on increase the number of points or error keeps on reducing.

(Refer Slide Time: 45:34)



Most important thing to note about the equation was that, it was a step differential equation and steps may show over quite height, was quite close to 1000, it was 1000.

So, that is epsilon, actually one over epsilon.

It is 1000, so the grid size is will matter moves one more important thing, we will observe that our solution is two parts.

So, if we take **the as** sir said, if we take the fast Fourier transformation, where will be two frequencies dominating, like he showed for the two modes; so, the nature, the analytical solution explains that part also.

So, we also took some smaller values of the grid for which the solution in diverge very rapidly, so that also about say limitation of a step size and thank you.

So, now, yes, I think again, now we missed one of the presentations, we will do it the **next class; you would get a proper...**

Excuse me.

Sir, in this, this the first graph.

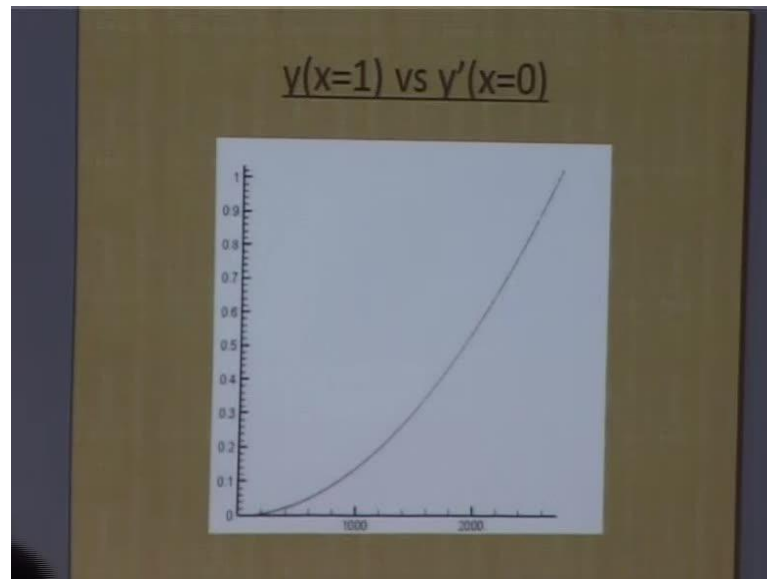
In this case, let me first graph.

(()).

Now, what is the question?

We saying like the graph is monotonic, is it necessary that it, (()).

(Refer Slide Time: 43:19)



It does not have to be, so see that point is what they have done is, what is called a grid search method, you identify a parameter space which define by y at x plus 1, and y prime, and then you keep on doing it, and then the moment you find out, where you are far field condition is satisfied there, you keep assuming in that area and take much more refined the grid search.

And this method may appeared to be more compute intensive, but there is a beauty to this method is that, sometimes you may have multiple values, at the wall, that could satisfy the far field condition; so, if you are doing shooting method, you always get only one of it.

So, you do not get the whole complement of values which can satisfy the far field condition. So, in the grid search method, you would get the whole set of it, so that is something that we could do.

But sir, suppose we have a higher value of y time, which is also satisfying you, while we doing that value but $(())$.

No, you will not that is what I have always, saying that you have to hire enough range, it is almost, like your bisection method, you know I mean.

$(())$ bisection method, if we come through and then make our epsilon smaller.

So, it is exactly the same it is not anything isotonic than that.

So, I, suppose the only thing that we did not hear, is anyone doing variable h calculation, but you see why I am saying this, because if you know happen to go to any of those can programs solvers.

What they have actually, they decide a point, the step size by local truncation error analysis. So, at each and every point you go there, you find out what is the optimum h for that particular step, so it is not like what shakthi said, that for the inner layer, you take one step size, and outer one, it is not necessary, because your method does not depend upon the fact, that you will have to have uniform spacing, you can give any spacing at any point in time.

So, that you could do, now only question that I wanted to answer was what Himanshu and Mohit showed that, even with 50000 points, there were some discrepancy on the seventh and eighth.

The answer is very simple, because all that you are doing, you are equating up to here. What happens to the next order, once you have exhausted this, the next one will play the role of leading error, so that is where the difference comes in, so that is where the difference would be, so there are no mysteries.

Are there any, if they have none, so I suppose, we now understand certain things, I did not discuss about various sources of error, but we have talked about one thing, quite a bit is the truncation error.

Today, we did talk about round off error and unfortunately this round off error does not get its due attention, but it is it could be very, very significant issue, especially when you are solving problems which are physically unstable.

When a problem is physically unstable, as a post to numerical instability, those physical instabilities are triggered by some background disturbances, that is where the noise environment place, the role in a computed solution; so, that is where the round off error does play a significant role, but it has not been very much well proved and analyzed, but slowly people are becoming aware of it, **lots of is the time**, for example, we do sum up these kind of work, where we find out that an aircraft is flying at an altitude of 11 kilometer, what kind of background disturbances we have, and how does it affect the flow field.

This kind of analysis is attempted, and when we do that, we realize that round off error actually plays a major role. So, what happens is, there are all kinds of strange prescriptions suggested. Few years ago I heard a France scientist actually took a patent for controlling round off error and which turned out to be total nonsense later.

So, this round off error is quite a significant parameter to consider to be important, when you are looking at the mean field, like equilibrium solution, like this we are getting. So, this is your equilibrium solution, the boundary layer you have observed, it is the equilibrium solution. Now, what one could do is, point out this equilibrium solution and see if those perturbations decay or amplify, that is how we do physical instability studies.

So, the perturbation field does play a great role, and that is where we should be paying particular attention to this. The one more thing, that I wanted to tell you is about this optimum, that we have seen couple of them, that still does not factoring one aspect, how does the error depend on f .

(Refer Slide Time: 53:11)

$$T^{n+1} = u^{n+1} - u(t^{n+1})$$

$$= h^4 \left(f_{nnn}^3 \left[\frac{w_2 a_{21}^3}{6} + \frac{w_3 (a_{31} + a_{32})^3}{6} - \frac{1}{24} \right] + f_{nn}^2 f_{nn} \left[\frac{w_3 a_{21}^2 a_{32}}{2} + w_3 a_{21} a_{32} (a_{31} + a_{32}) - \frac{1}{6} \right] + f_{nn}^3 \left[\frac{1}{24} \right] \right)$$

We have been totally silent, we have been playing around with the coefficients of the RK3 parameter, and we are talking about how to minimize make these two component 0, but still this remains.

(Refer Slide Time: 53:32)

$$\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} = 0$$

$$\frac{\partial u}{\partial t} = - \underbrace{c \frac{\partial u}{\partial x}}_f$$

$$y_e = \frac{1}{e^{-1} - e^{-1000}} (e^{-t} - e^{-1000t})$$

$$E = \frac{\sqrt{\sum_{n=1}^N (y_e - y_n)^2}}{y_e}$$

Suppose, I give you a problem, where f is defined I could do that, for example, this is what you can think of negotiating in their future that, this is what we keep talking about. So, if I look at it, I could write it like this.

So, this could be f , that could be the f , that we need to solve for this particular problem, then what happens, then you should then minimize the error, not only these two terms, but all the three terms taken together, that would be a much more of a global search global minimum.

And well as you can see, it is coming away on as the third assignment, you would like to do, so you can actually build up on your knowledge, what you have done on first assignment, use it on the third, and come back and share your experience.

So, we will do that, so that is something, which something we have done. Now, another aspect of optimization that we talked about, it is sort of lop sided approach, because we kept our attention focused only on truncation error.

Is that the only source of error, apart from truncation and round off, we have already seen in our error analysis part of the discussion. There are physical sources of error, which relates to we have seen numerical in stability numerical, stability also can cause error. We have seen if there are convective solutions, if they are going at the right speed, the group velocity is major parameter dispersion.

So, you can see there are other aspects of optimization, I probably have mentioned to some of you, we do some such work, one of our PhD student is actually developing a RK3 scheme, it is a PhD program, so it what you people are doing it is not trivial; so, people do look at corresponding thing, but there we look at all those physical mechanisms, how to minimize those kind of errors.

So, if we have a wave propagation problem, how do we minimize the error there; so, that is also something, that we can do so, and just go, why, I, suppose, now with this exercise, all of you understand that how one does research.

It is not always something, that you reproduce, that is already there in the text book, there are issues like as I said George Howell's quotation and issues within issues, so people when you go deeper into it; then, you see there are lots of other aspects emerging.

I think we had a very faithful discussion, today unfortunately missed one that we will catch up with it, and any of you, well rest of you, if you want to come up or discuss, please do it either on the podium or we can do it through the email.

We have a course ID, we can bound ideas from one on each other, and I did not want to have this chat group or something I think that ends up becoming chat, it is better to keep away from all this fancy things.

Well, I think with this, I would now say that, we have mostly done, if any of you have any questions about this or anything else feel free to discuss, **we have**, now I have a 45 minutes to discuss if you want to.

$\frac{1}{n}$ is noted error by n versus, the n which can be misleading, because the error can be a constant and $\frac{1}{n^2}$.

Which one you are talking about elliptic.

In the last presentation.

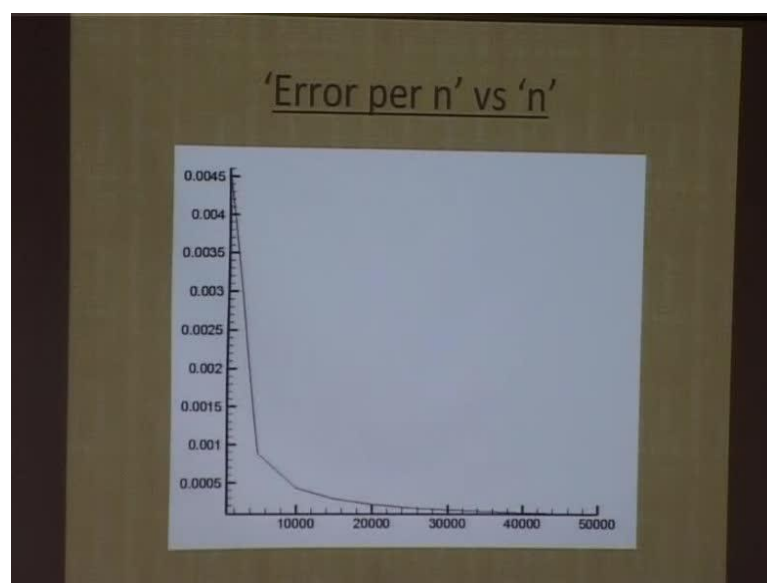
Oh, that shakthi's presentation.

Yes.

Let me, what is your point Manish.

$\frac{1}{n}$, starting error by n versus n .

(Refer Slide Time: 43:19)



This one, yes, so it is a basically sort of a average error integrated over the domain, you are added the error for all the end points and then.

What is error versus n be a better measure of that, because this can be illustrating.

This is always, well that is why statistics is not pure science, when you average out you can put your hand in a hot stove and chill at a, I mean, you can say a normal experience that is not true, so that is a problem with always statistics, so be very cautious, when you deal with statistics.

Do you know that, there are all on a lighter node, **we are**, we have some time to talk about people are found out a very strong correlation with boldness with medicate support in USA.

So, of course, the people who are bald, they are older, and they also go to the doctor, but is that a good correlation to draw.

So, **I** suppose statistics can be very misleading, so enough, of course, you drop skills, whenever you talk about averages, you are talking about only one statistics, what about the higher statistics; so, RMS error is one plus the second orders statistics well.

If you are fancy, you can go to kurtosiskun us all kinds of things, third order, fourth order statistics and find out.

Excuse me sir, in that notation error, we said there are three methods of actually minimizing the error, **so one is**, if within the coefficient is 0, other is the differentiation, what is the third one sir.

No, I said that, if f is prescribed, then I could do a global search and I did not say 3, I did add few more, that if you are interested in optimizing certain aspects of error, like that is what we did we corrected the Frauen Namen analysis from that angle.