Computational Techniques

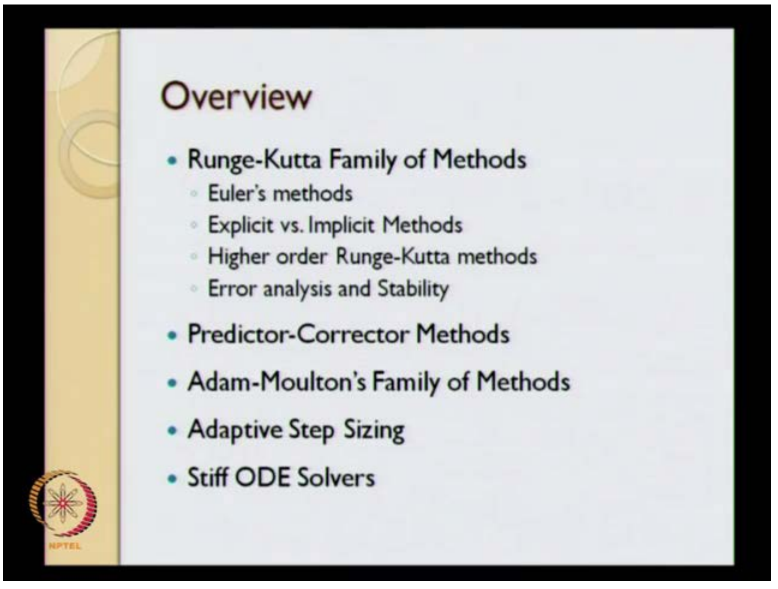
Prof. Dr. Niket Kaisare
Department of Chemical Engineering
Indian Institute of Technology, Madras

Module No. # 07

Lecture No. # 07

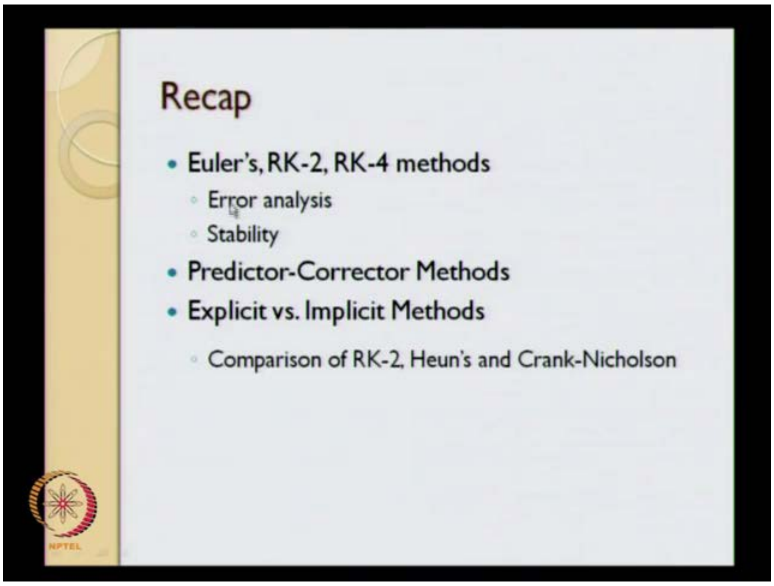
Ordinary Differential Equations (Initial Value Problem)

(Refer Slide Time: 00:34)



Hi and welcome to this lecture seven of module seven. In the previous lectures, what we have been discussing is ordinary differential equation, initial value problem and we have considered several methods error analysis and stability analysis for these methods. So, I will just go over and do a very quick recap on power point slide and tell you what we are going to cover in the next two lectures.

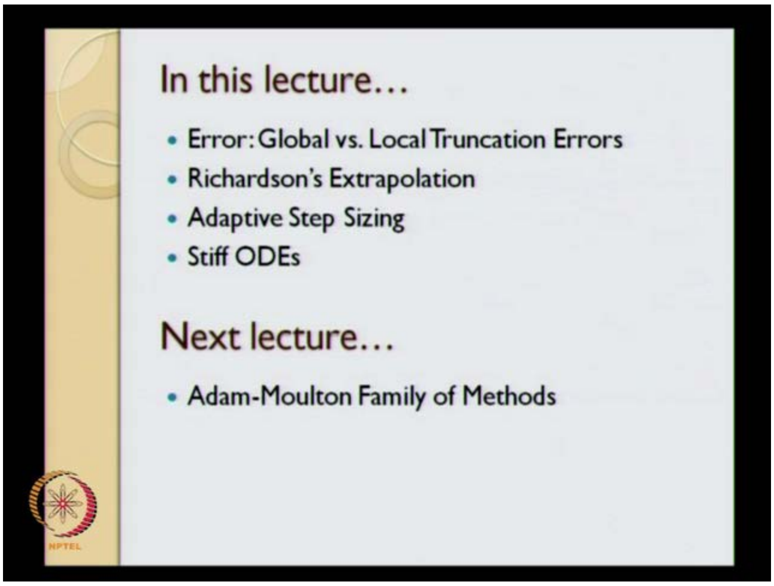
(Refer Slide Time: 00:37)



What we have covered so far is ,we started off with Euler's method, then we talked about in general, about Runge-Kutta methods; made derivation for Runge-Kutta second order method as well as solve with the results for, fourth order Runge-Kutta method we performed error analysis, and saw that the truncation error for Euler's method was the accuracy is of the order x square for the r k two method. The accuracy is of order of h cubed for r k four method. The order accuracy is of the order of h to the power five. All these accuracies that are errors that we talked about or in context of what is known as local truncation error and I will come back to that in the next slide, so we did a error analysis for the Runge-Kutta family of methods followed by the stability analysis. Then we took up Heun's method and put in a predictor-corrector form as well as. we considered a explicit versus implicit methods specifically we took the explicit Euler's method and implicit Euler's method and saw the stability results for a linear system.

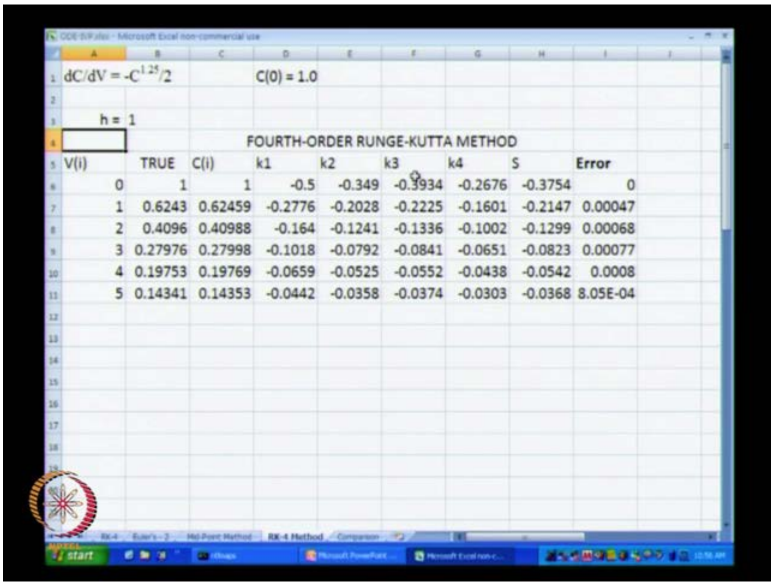
We compared the second order r k method with the Heun's method, which is a predictor-corrector method and compared with the Crank-Nicholson, method which is a semi-implicit method. So, and the final thing we covered in the previous lecture was extension to multivariate case. A specifically we saw the Euler's methods extension to the multivariate case and then we discussed, how you can extend in in the similar manner r k two and r k four methods, as well those extensions are actually not just limited to r k type of methods the same extension will work in any other type of method for o d e solving that we are going to consider so far ok.

(Refer Slide Time: 02:33)



So, that was what we have done so far in the first six lectures, in the second, in the next two lectures of this series, this is what we are going to consider. In this particular lecture, we are going to compare global versus local truncation error and then we are going to cover three advanced topics in this lectures, I will go over I Chardson's extrapolation recall that Richardson extrapolation, something that we had done, when we were discussing integration techniques. So, what we do is we will get h to the power nth accuracy solution and use Richardson extrapolation to get h to the power n plus one accurate solution that is what we had seen in module six of this particular course and how it would be extended to o d e initial value problem we will talk about that today, more importantly I am going to talk about what it means by adaptive step sizing the similar kind of idea that is used in Richardson's extrapolation a slight modification of that would be actually used in deciding the step size in the adaptive step sizing method and finally, we are going to consider what is known as stiff o d e's I will essential define what is stiff o d e is means and why the stiff o d e's cause numerical problems and this is what we are going to cover. And finally, in the next lecture we are going to cover Adam-Moulton family of methods and that will essentially finish what I intent to do in o d e initial value problem, module seven ok

(Refer Slide Time: 04:15)



$dC/dV = -C^{1.25}/2$ C(0) = 1.0

h = 1

FOURTH-ORDER RUNGE-KUTTA METHOD

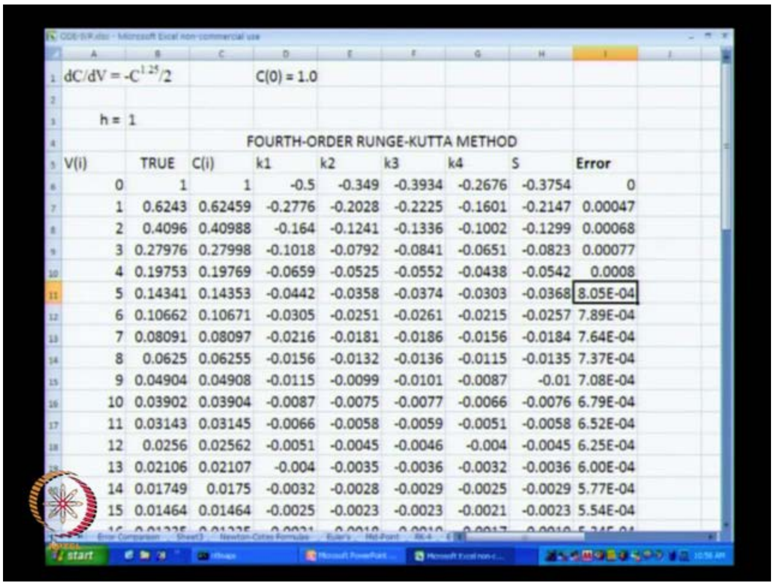
V(i)	TRUE	C(i)	k1	k2	k3	k4	S	Error
0	1	1	-0.5	-0.349	-0.3934	-0.2676	-0.3754	0
1	0.6243	0.62459	-0.2776	-0.2028	-0.2225	-0.1601	-0.2147	0.00047
2	0.4096	0.40988	-0.164	-0.1241	-0.1336	-0.1002	-0.1299	0.00068
3	0.27976	0.27998	-0.1018	-0.0792	-0.0841	-0.0651	-0.0823	0.00077
4	0.19753	0.19769	-0.0659	-0.0525	-0.0552	-0.0438	-0.0542	0.0008
5	0.14341	0.14353	-0.0442	-0.0358	-0.0374	-0.0303	-0.0368	8.05E-04

So, now, let us go to this lecture, what i will do is i will again go to Microsoft excel and pick up a sheet that we were using in the previous lecture. So, let me just bring up Microsoft excel over here.

So, this is the sheet that we were using in the previous lecture, what we wanted to solve is d c by b v equal to minus c to the power 1.25 divided by 2 this expression, we obtained by were deriving the design equation for a plug flow reactor, we start with initial concentration of the desired of the reactant as equal to one and starting with this initial concentration of one and the actual concentration keeps decreasing as the time, as the volume keeps increasing.

So, we had what we had done is - we had derive or we had used this particular method for various different values of h for h equal to 0.125 we had use this method and what we had done is at volume equal to 5, we saw that the error was ten to the power minus seven.

(Refer Slide Time: 05:28)



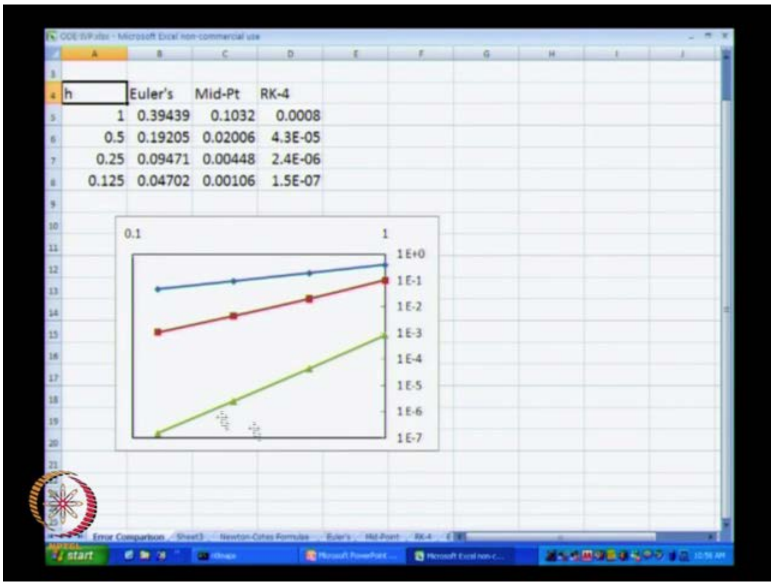
$dC/dV = -C^{1.25}/2$ C(0) = 1.0

h = 1

FOURTH-ORDER RUNGE-KUTTA METHOD

V(i)	TRUE	C(i)	k1	k2	k3	k4	S	Error
0	1	1	-0.5	-0.349	-0.3934	-0.2676	-0.3754	0
1	0.6243	0.62459	-0.2776	-0.2028	-0.2225	-0.1601	-0.2147	0.00047
2	0.4096	0.40988	-0.164	-0.1241	-0.1336	-0.1002	-0.1299	0.00068
3	0.27976	0.27998	-0.1018	-0.0792	-0.0841	-0.0651	-0.0823	0.00077
4	0.19753	0.19769	-0.0659	-0.0525	-0.0552	-0.0438	-0.0542	0.0008
5	0.14341	0.14353	-0.0442	-0.0358	-0.0374	-0.0303	-0.0368	8.05E-04
6	0.10662	0.10671	-0.0305	-0.0251	-0.0261	-0.0215	-0.0257	7.89E-04
7	0.08091	0.08097	-0.0216	-0.0181	-0.0186	-0.0156	-0.0184	7.64E-04
8	0.0625	0.06255	-0.0156	-0.0132	-0.0136	-0.0115	-0.0135	7.37E-04
9	0.04904	0.04908	-0.0115	-0.0099	-0.0101	-0.0087	-0.01	7.08E-04
10	0.03902	0.03904	-0.0087	-0.0075	-0.0077	-0.0066	-0.0076	6.79E-04
11	0.03143	0.03145	-0.0066	-0.0058	-0.0059	-0.0051	-0.0058	6.52E-04
12	0.0256	0.02562	-0.0051	-0.0045	-0.0046	-0.004	-0.0045	6.25E-04
13	0.02106	0.02107	-0.004	-0.0035	-0.0036	-0.0032	-0.0036	6.00E-04
14	0.01749	0.0175	-0.0032	-0.0028	-0.0029	-0.0025	-0.0029	5.77E-04
15	0.01464	0.01464	-0.0025	-0.0023	-0.0023	-0.0021	-0.0023	5.54E-04

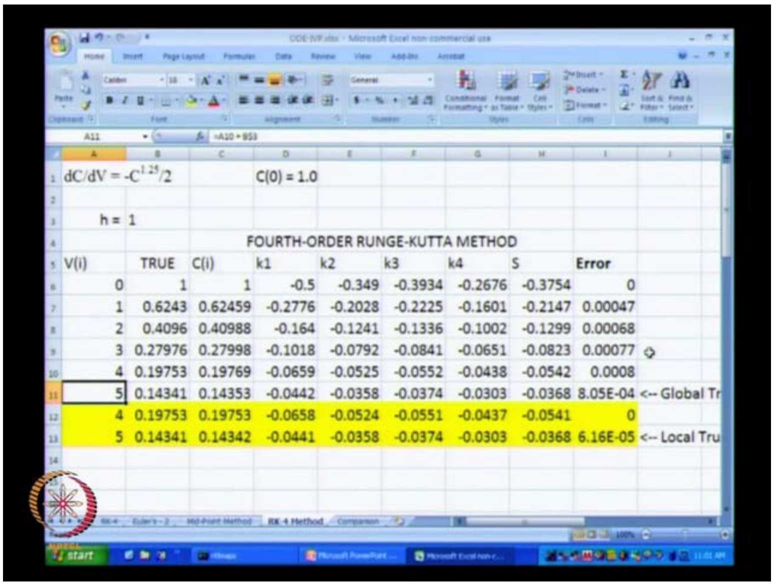
(Refer Slide Time: 05:40)



h	Euler's	Mid-Pt	RK-4
1	0.39439	0.1032	0.0008
0.5	0.19205	0.02006	4.3E-05
0.25	0.09471	0.00448	2.4E-06
0.125	0.04702	0.00106	1.5E-07

What we did next was, we change h to h equal to 1 and for h equal to 1, the error in case of volume v equal to 5 was equal to10 to the power was of the order of 10 to the power minus 4 and with all these results, we had made error comparison between Euler's midpoint and r k 4 method and this error comparison is shown on this particular graph, all this we had done in two lectures earlier.

(Refer Slide Time: 05:55)



$dC/dV = -C^{1.25}/2$ C(0) = 1.0

h = 1

FOURTH-ORDER RUNGE-KUTTA METHOD

V(i)	TRUE	C(i)	k1	k2	k3	k4	S	Error	
0	1	1	-0.5	-0.349	-0.3934	-0.2676	-0.3754	0	
1	0.6243	0.62459	-0.2776	-0.2028	-0.2225	-0.1601	-0.2147	0.00047	
2	0.4096	0.40988	-0.164	-0.1241	-0.1336	-0.1002	-0.1299	0.00068	
3	0.27976	0.27998	-0.1018	-0.0792	-0.0841	-0.0651	-0.0823	0.00077	
4	0.19753	0.19769	-0.0659	-0.0525	-0.0552	-0.0438	-0.0542	0.0008	
5	0.14341	0.14353	-0.0442	-0.0358	-0.0374	-0.0303	-0.0368	8.05E-04	<-- Global Tr
4	0.19753	0.19753	-0.0658	-0.0524	-0.0551	-0.0437	-0.0541	0	
5	0.14341	0.14342	-0.0441	-0.0358	-0.0374	-0.0303	-0.0368	6.16E-05	<-- Local Tru

So, now, what I am going to do is I am going to talk about, what is known as local and global truncation errors. So, we do not need any of this information. So, I will just delete that and I will just keep the values of v from zero to 5. So, what we are did over here is we started with the initial condition at v equal to zero and reached v equal to 5 over here. Now, what I am going to do is I am going to take these two rows and then just paste them over here and I will restart them I will just color them with a different color. So, we are able to see how these are actually going to be different.

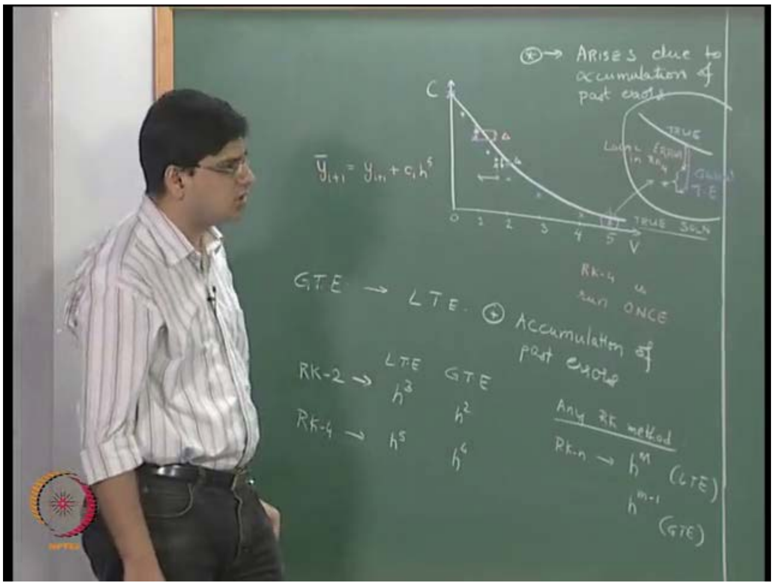
So, I will just fill them with a yellow color over here and I will redo this, calculations for four equal to 4 and 5 and instead of using the approximate value over here. What I am going to do is - am going to use the real value of the true value of c i plus one that we get am going to use that over here. And if i were to use the actual true value so, I will just copy this particular true value and paste that true value over here by putting paste values.

Now, what we see over here is let us say, we started off from v equal to zero and we reached v equal to 5 when we did that the error was 6 into 10 to the power minus 4. However, were we to start with the exact value of at v equal to 4 and then take one step to go on to v equal to 5, we will find that the error is significantly lower in this particular case. This error that we get is essentially, the local truncation error and this error, that we have got in is the global truncation error.

So, the difference is like this, in the local truncation error, what we do is - we assume the previous value, that we had was the actual true value. So, the value at v equal to 4 was true value, if the value at v equal to 4 had been the true value, what would be the error at v equal to 4 plus h that is v equal to 5. So, if we started off with the true value, we will reach this particular value at v equal to 5 and in that particular case, the difference between the true value at v equal to 5 and the numerically computed value at v equal to 5 is 6 e minus 5. However, if we were start at v equal to zero and reach v equal to 5 through 5 steps in that case, the global truncation error is going to be 8 into 10 to the power minus 4 which is an order of magnitude higher than the local truncation error that we get over here.

So, that is the difference, really between the global truncation error and the local truncation error. So, I will just go back to the board and discuss a little bit more about global and local truncation errors I would not do the derivation, but essentially give you the results for the Runge-Kutta family of methods.

(Refer Slide Time: 10:04)



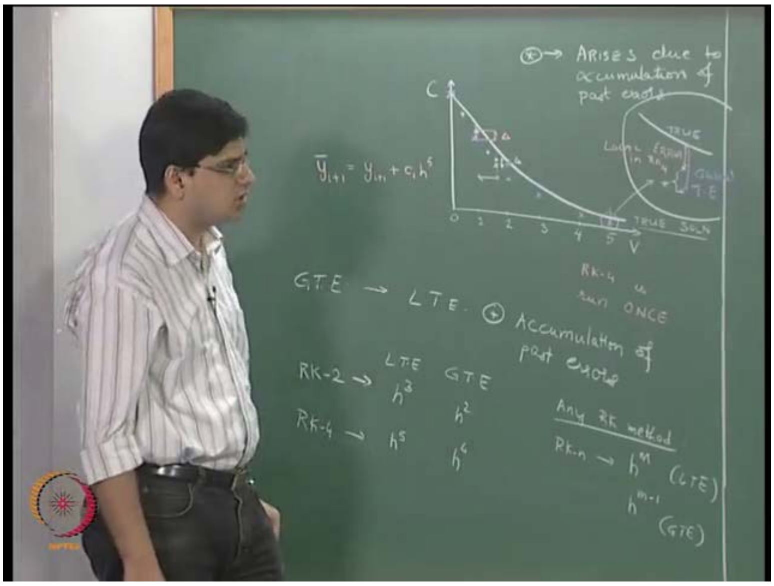
So, what we saw in the excel sheet right now, just now is this, we wanted to compute the value of concentration, how the value of concentration changes with the volume and we were interested in calculating, the value of concentration at volume v equal to 5 and am going to blow up the differences the between the numerical value and the actual value, just so that we are able to see the what we really mean by the local and global truncation errors. So, what we started off? We started off at v equal to 1, and the white line that I am showing over here; let that represent the true solution.

So, we started off at v equal to 1 and reach, sorry, v equal to zero and reached v equal to 5 in 5 steps 1,2,3,4 and 5. So, this is how the true value goes. Now, let us say that we use a numerical method r k four type of a numerical method. In order to, compute the same values at 0, 1,2,3,4 and 5. Now, we are starting off again at the same concentration c equal to 1 at volume v equal to zero. We use the chosen method, say the r k four method once and form this particular point let us say, that we reach this point over here, we apply the same r k four method. Once again and from this point we will reach this point to that i am showing over here from this point onwards, if i apply this method again I will reach here, I will from this point onwards, I will reach here and finally, I reach this particular point.

So, if we take a magnifying glass and just blow this thing up, what we will see is we will the curve coming like this and this purple cross over here. So, this is the magnified view of this particular region and this is the true solution. So, what we do next is let say, some how this particular solution is available to us, if this solution is available to us we restart our r k four method at v equal to four we restart out r k four method at the true value. So, if we restart this, r k four method at the true value and run this r k four method once, then from this point the next point that we get the next numerical solution ends up being over here this particular dot that I have shown. And when we blow up this particular picture what we get is really that this dot ends up over here if we had started with the true value.

So, now, if we were to calculate the errors in getting this purple dot what this represents, this particular gap, represents the local error in r k four, what that means is that the y i plus1. The true value of y i plus 1 is equal to y i plus 1 from the r k four method plus an error term, which we can write it as some concern c 1 h to the power 5.

(Refer Slide Time: 10:04)



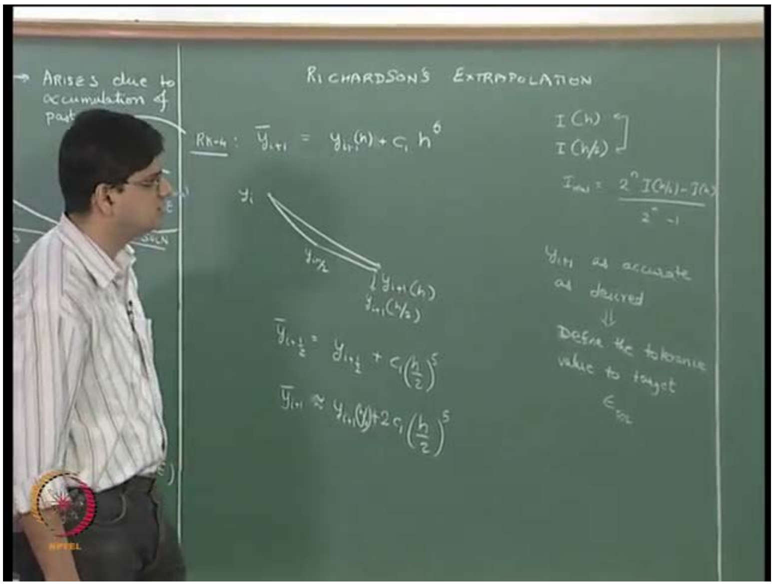
So, this error is proportional to the fifth power of the step size that I have chosen. Now, this error is the local truncation error. This error is the error, going from the ith value to i plus 1th value assuming the ith value was the true value. So, this represents the local error. Now, when we started with c e c at v equal to zero, we went from c c zero to c 1, when we went from c zero to c 1, there was a local truncation error associated with this value of c 1. So, when we then go from this value, this numerical value to the next numerical value, there will be some error associated with the fact that the numerical method is not accurate. This has certain error itself, there will be some other error associated because of propagation of this particular error.

So, this error results in some of the error in the future step and there will be some error that will be cause because the numerical method itself is inaccurate. So, the effect of both these errors is comes in the global truncation error and this difference is going to be the global truncation error. And this particular difference between the global truncation error and the local truncation error, this arises due to accumulation of past error. So, the global truncation error, we can say is going to be some kind of a combination of the local truncation error plus accumulation of past error, I am, I do not mean this as an equation, I just mean this as a literal statement of fact is that the global truncation error has a component of the local truncation error. As well as component, because of the propagation of the errors go starting at starting point and going to the desired point as for this numerical method.

In general, what we had seen is for r k 2 method the local truncation error was of the order h to the power 3, the global truncation error will be of the order h to the power 2 for r k four method. The local truncation error was h to the power 5 the global truncation error would be of h to the power 4. So, for any r k method, what we actually get is for an r k r k n method, if the accuracy is h to the power m in the local truncation error, then the accuracy is going to be h to the power m minus one in terms of global truncation error.

So, the forth order r k method is going to be h to the power four accurate in the global sense and not h to the power 5 accurate, as we had derive in the previous lectures, what we had derived in the previous lectures is local truncation error, which does not consider the fact that as we propagate into the future or into the larger volumes, this the effect of truncation error, the effect of truncation error that have been accumulated in the past also effects the overall error in the system. So, that is the difference mainly between the global truncation error and the local truncation error.

(Refer Slide Time: 09:15)



So, that is one part that I wanted to cover today. The next thing, I will cover today is will go on to a little bit more advance topic, really what I want to cover is adaptive step sizing but, before going to adaptive step sizing. we will look at Richardson's extrapolation you will recall, what we had done in Richardson's extrapolation in module six was we computed the integral for h, we recomputed the integral for h by two and used these two integrals in order to compute a better value or a more accurate value of integral.

So, and we had said, I knew was going to be equal to 2 to the power n multiplied by i h by 2 minus i h divided by 2 to the power n minus 1. So, this is what we had derived in the previous module, we are essentially going to go, over all most exact same derivation for Richardson's extrapolation in case of the Runge-Kutta methods or in case of, o d e solution methods. So, the idea is this, what we want to do is - we want to get y i plus 1, as accurate as desired, what do we mean by as accurate as desired is that we need to define what tolerance value that we need.

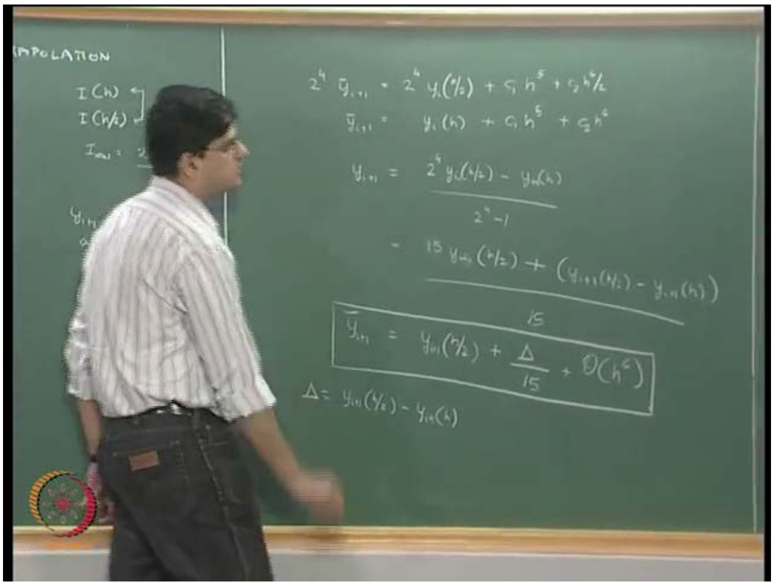
So, we need to define and we will call this, as if we have been doing in the past, will call this as epsilon tol - that is - the desired tolerance value or desired error value that we will give. Now, in Richardson's extrapolation before going to and or talking about this particular tolerance values and so on will go over there is Richardson's extrapolation the idea is this the true value y i plus 1 the true value is going to be equal to y i that is complete or y i plus 1, which is computed numerically plus some constant c 1 multiplied by h to the power 5 for the r k 4 method.

This is what essentially, we had written over here, I have just rewritten this particular equation over here for r k 4, we have the true value y i plus 1 bar equal to y i plus 1 plus c 1 h to the power 5. What I will do is I will just modify this a little bit in line with what we had done over here. And I will put in bracket y i plus 1 h plus c 1 h to the power 5. So, this is using the r k 4 method. Once going from y i going from y i to y i plus 1. This is what we get. Now, let us, redo the same thing going from y i to y i plus 1 in 2 steps. In this case, the step size is going to be h by 2. So, we will go from y i to y i plus half and from y i plus half, we will go to y i plus 1 computed at with a step size of h by 2.

So, this particular method, when we use this method is going to be more accurate, as we have been seeing all throughout this method is going to be more accurate in giving us the value of y i plus 1 compare to the method in which we are going to use one single step to go from y i to y i plus 1. Now, there will be error that is going to be associated with this particular step, as well as an error that is associated with this particular step. So, we will have our y i plus half bar is going to be equal to y i plus half plus c 1 h by 2 to the power 5. So, this is going to be our true value of y i plus half y i plus 1 bar is going to be equal to that the value computed at y i plus 1 from this particular guy plus the error that that we will get at y i plus 1.

So, now, this error is also going to propagate into the future. So, without going actually into any of the numerical complexities into this, I will write this y i plus 1 is going to be approximately of the form of y i plus 1 plus 2 times c 1 h by 2 to the power 5, the first c one h by 2 to the power 5 comes, because of the go from y i to y i plus half the second c one h h by 2 to the power 5 comes because of the local truncation error going from y i plus 1 to y i plus half to y i plus 1.

(Refer Slide Time: 25:42)

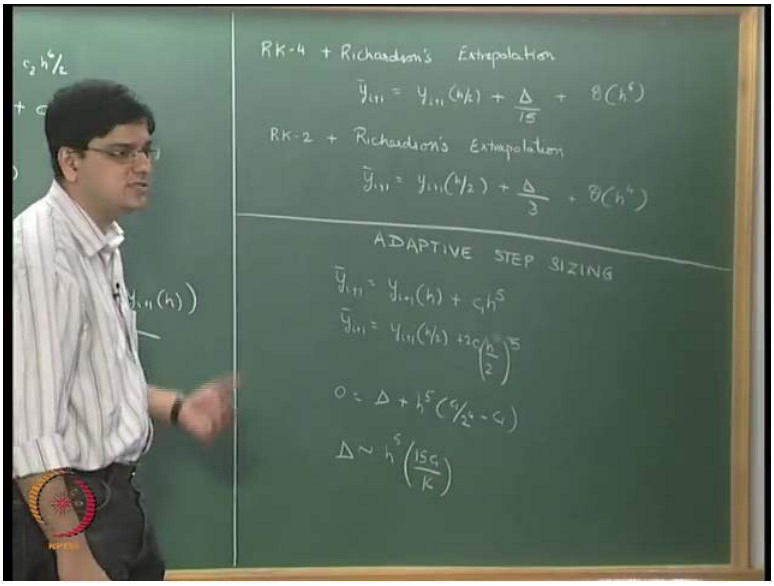


So, this is essentially, how we will write our y i plus 1, we when we go for the true value of y i plus 1 related to y i plus 1 computed by taking two steps and because it is computed by taking two steps, I will put in the bracket y i plus 1 h by 2, as I have we had done before we will multiply, this particular equation by 2 to the power 4 not 2 to the power 5 but, 2 to the power 4, just as we had done in the integration method we will multiplied by 2 to the 4 and then subtract this equation from this equation. So, we will get 2 to the power four y i plus 1 equal to 2 to the power 4 multiplied by y i h by 2 plus c 1 h to the power 5.

So, we had two times c 1, we had two times c 1 h by 2 to the power 5, when we multiplied this with 2 to the power 4 all this got canceled away. And we were left with c 1 h to the power 5 and that is what I have written over here as well as y i plus 1 i forgot the bar over here is going to be y i computed with h plus c 1 h to the power 5 and of course, there are c 2 h to the power six is c 3 h to the power 7 terms also and so on over here which are not going to get canceled, because we will have we will essentially have plus c 2 h to the power 6 by 2 and here we will have c 2 h to the power 6. This is the term that I had not written down earlier.

And now, when we subtract these two equations and then divide throughout by 2 to the power 4 minus 1, we are going to get y i plus 1 is going to be equal to 2 to the power four y i h by 2 minus y i h divide by 2 to the power 4 minus 1. So, this is essentially, what we are going to get we can write this as sixteen times y i plus 1 I should have here sixteen times y i plus 1 h by 2, I can write this as fifteen times y i plus 1 h by 2 minus y or sorry plus y i plus 1 h by 2 minus y i plus 1 h divided by 15 and which we will be able to write it as y i plus 1 computed at h by 2 plus delta divided by 15. Where delta is defined as the difference y i plus h by 2 minus y i plus 1 h and this is going to be our final result and similar derivations can be done for the r k 2 method also and for r k 2 method we are going to get.

(Refer Slide Time: 29:13)



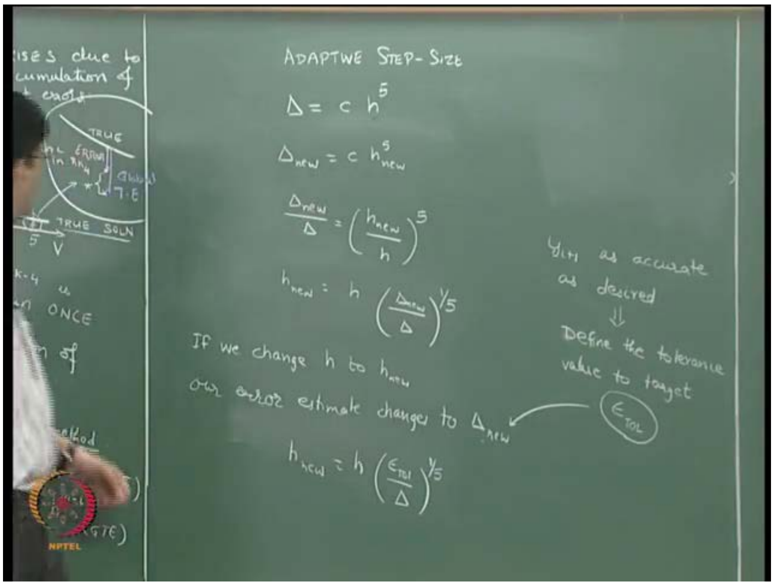
So, for r k 4 ((no audio 29:13 to 30:22)) now, how this 15 comes in, it comes in 2 to the power four minus 1 likewise, this 3 comes in because it is going 2 to the power 2 minus 1 the exponent is the order of accuracy minus 1. The local truncation error is h to the power 5 accurate over here h to the power three accurate over here. So, the exponent for two is going to be one less than the local truncation error accuracy. So, it is two the power 2 minus one that that becomes three and the overall accuracy of this method increases by one order. So, it is increased from h cubed to h to the power 4 and this is for the r k 2 method this is for the r k 4 methods that we get.

So, these are the results of Richardson's extrapolation, now the idea is this particular difference delta this in Richardson's extrapolation was use in order to improve the accuracy of the r k method and go to a more accurate method however this delta says something about the true error that we might expect to have in this particular system; to the new topic, we are going to tackle is adaptive step sizing.

So, y i plus 1 bar is y i plus 1 computed with h plus c 1 h to the power 5 y i plus 1 bar is going to be equal to y i plus 1 computed at h by 2 plus 2 times c 1 h to the power 5 divided by h, sorry, h by two to the power 5. So, when we take the difference between these two, we will get zero is equal to delta plus h to the power 5 c 1 by 2 to the power four minus c1. So, what we have done is - we are subtracted this equation from this equation, these two are the true values and therefore, the differences going to be zero the difference between these two values is going to be equal to delta and the difference between these two values is going to be some constant multiplied by h to the power 5.

So, we take this particular equation on to the other side and we can, so this is going to be 1 by 16 minus 1 take it to the other side will become 1 minus 1 by 16 or 15 by 16. So, we can write this as delta is going to be the estimate of the delta is going to be h to the power 5 multiplied by 15 c 1 by 16 or it is some constant this of this particular form this particular value is constant and it depends essentially on y i and y i plus 1 based on the mean value theorem and the exact value we will not have an estimate of unless the true solution is exactly known.

(Refer Slide Time: 34:01)



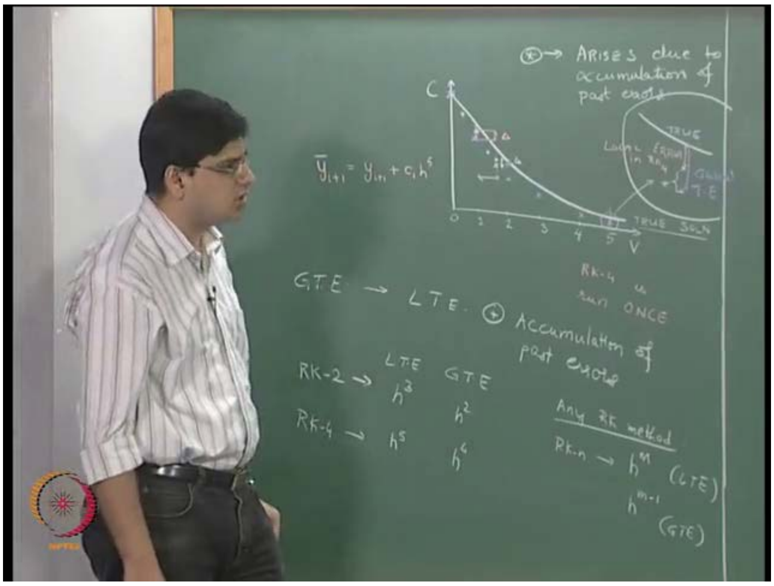
But what this says is that the delta that is the difference between these two guys is proportional to h to the power 5. So, delta is some constant some other constant c multiplied by h to the power 5 c could be 15 c 1 divided by 16, if you want to write it that way.

So, this is the value of delta this is estimate value estimate of this particular difference delta, now if h was the step size, this is going to be the delta that will get, so delta at new for another h new is also going to be c multiplied by h new to the power 5. So, if we were to change h from the actual value of h to a new value of h h new this delta will change from this true delta to the ah delta new.

Now, we can take basically divide this particular equation by this equation and what we are going to get is delta new divided by delta is h new divided by h to the power 5 and we can just rearrange this and we can write this as h new equal to h multiplied by delta new divided by delta to the power 1 by 5. So, what that what this equation says is... So, if we change from h to h new, the difference between a more accurate method and the less accurate method is going to change from delta to delta new.

So, now, if we instead of choosing this the delta new arbitrarily, if we choose that delta new equal to the tolerance value, then the h new that we are going to get is the step size that will help us in reaching that particular tolerance value.

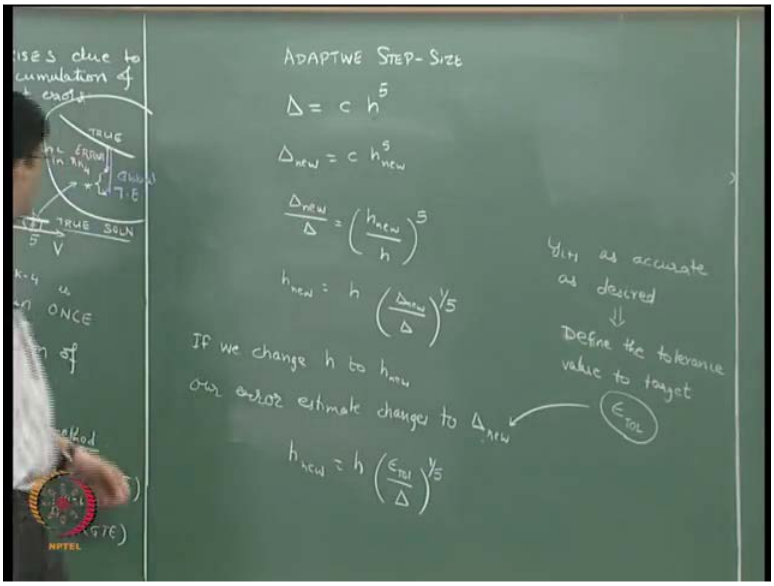
(Refer Slide Time: 37:14)



So, what we are going to write over here, is that h new is going to be h multiplied by epsilon tolerance divided by delta to the power 1 by 5. So, what we mean by that is we can go back to this particular schematic that we had drawn for the r k 4 method is if this particular error is not acceptable to us.

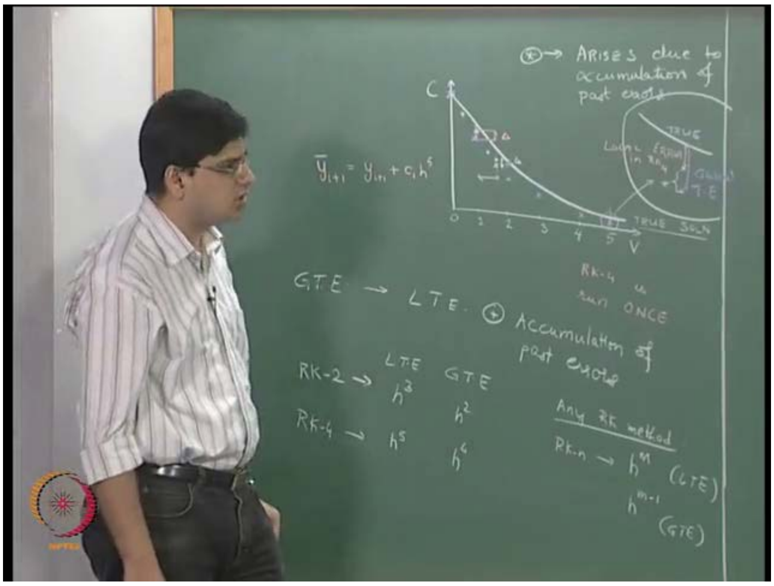
What we need first is we do not know the true solution. So, we need a method to find out to estimate, what this particular error is going to be, so what we will do is in addition to running this particular method, using this step size we will also run this method, using the h by 2 step size and we will get this red crosses using h by 2 this guy is delta.

(Refer Slide Time: 38:10)

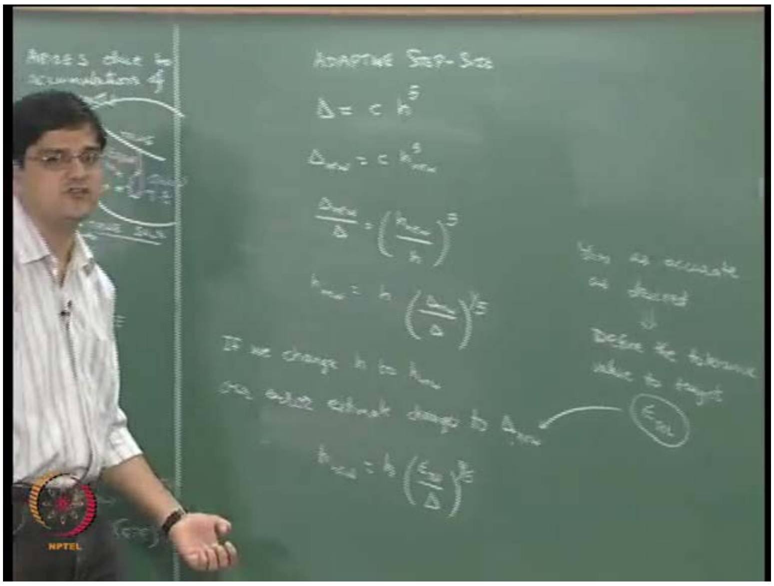


Now, if this delta is acceptable to us we keep the same h, if this delta is not acceptable to us we are going to change that. How we are going to change that h? We are going to scale that h in this particular manner, it depends on the tolerance value divided by delta to the power 1 by 5 that 5 comes in because, this method is accurate to the order of h to the power 5.

(Refer Slide Time: 38:26)

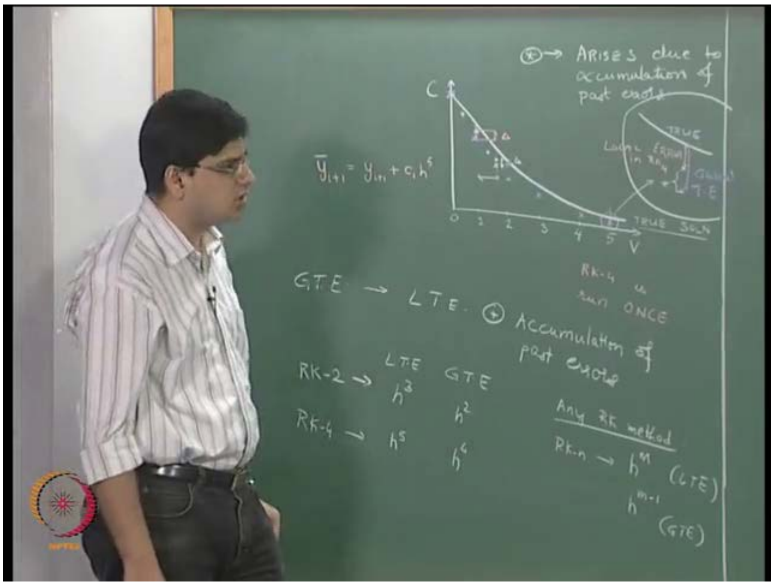


(Refer Slide Time: 38:36)



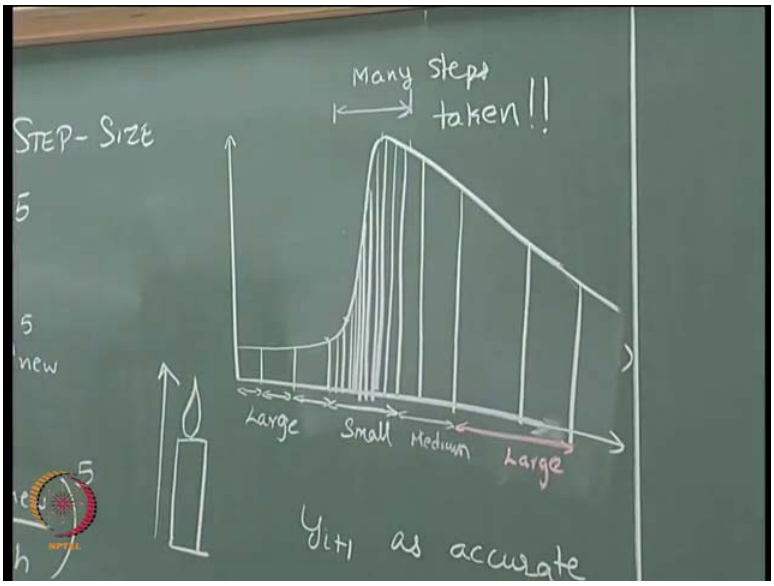
So, if this delta is too large, what we are actually going to do is we will shrink the step size if the delta is much larger than the epsilon tolerance, if delta is larger than the epsilon tolerance, what is going to happen is this number is going to be less than one, if this number is less than one h is going to shrink a little bit.

(Refer Slide Time: 38:48)



So, what is going to happen next is instead of taking a full step h, of this form we will then decide to take a shorter step h. So, instead of going all the way over here, we will decide our h is going to be this much and we will what we will do is from this point to this point, we will take we will want to take a step of h, again we repeat this adaptive step sizing once again. What we will do is we will take this particular h and redo this by taking two steps and those two steps are shown as circles; again we will compute this delta, again if this delta is not good enough, we will take a smaller step than the step that we had taken previously as a result of this, we will keep adapting the steps, as we go on further along this particular curve.

(Refer Slide Time: 39:52)



This is what is going to be an adaptive step sizing, where is adaptive step sizing useful in lot of example, let me just give you one particular example, where it is going to be useful, if this is that particular device, there will be a flame that stands over here. So, it could be a land lighter it could be a candle and if you start plotting the temperatures along this particular, direction you will get the temperatures to be fairly cool over here and then the temperature will start raising and then the temperature will shoot up; and the temperature will again fall, because there is a lot of heat loss taking place beyond this flame for example, if this is candle and we put hand over here, we do not feel too much heat of candle, because the heat has all dissipated throughout.

So, now, if we want to make a model for the system and then run an r k 4 method with adaptive step sizing on a curve of this sort, what happens is that this the slope is very low in this particular region. So, in this particular region what the adaptive step sizing will do is it will take fairly large steps. So, it might take possibly, say three steps to reach over here. Now, a lot is happening in this particular region, as a result of this, the things are changing quite rapidly. So, the slope at this point is drastically different than slope at this point. So, if we are going to keep using the same large step sizes in this region also, we will miss the overall physical characteristics, that we see over as a result the adaptive step sizing is going to take multiple number of steps in this particular region.

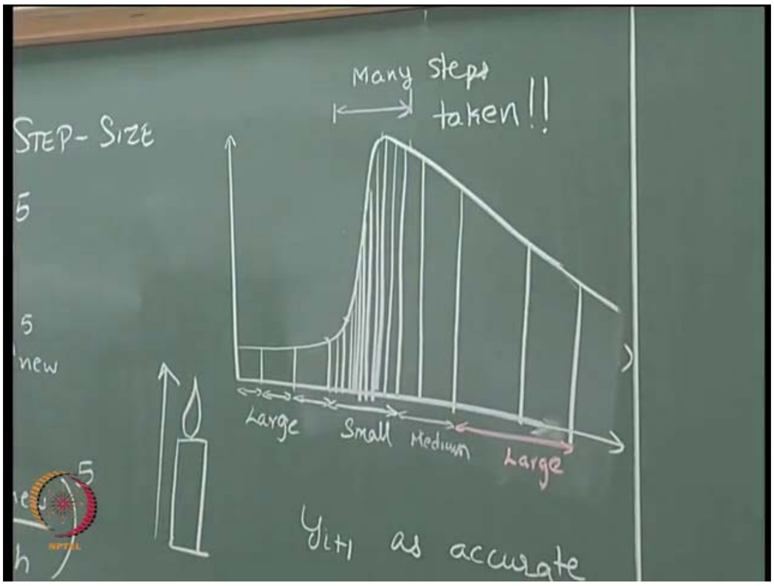
Now, the overall curve is tending to becomes smoother, because the curve tends to become smoother, the step sizes will increase. And this is what we, this is how we were going to get if we were to use adaptive step sizing. So, h over here are large in this region h s are small in this region h sizes will be medium. And finally, in this region the sizes of h will be large again.

So, what happens is in the region, where a lot of things are gradients are changing quite rapidly, you have many more steps taken and this is what the adaptive step sizing method is going to do for our system. So, that is essentially about the adaptive step sizing. So, let us consider dy 1 by dt equal to minus 1000 multiplied by y 1, what this means is that this particular lambda value is very large. As a result, this system is going to respond very quickly for example, if we were to plot y one against p, it is going to respond in this fashion, the time scale for the time scale tau is going to be 1 divided by lambda. So, the time scale is going to be equal to 10 to the power minus 3. So, in approximately 5 milliseconds, the system would have gone from whatever initial condition it was to zero in about 5 milliseconds. So, this is going to be zero, this is going to be .001, .002 and so on, and this is .005.

So, this is what this particular system is going to respond. Now, let us consider another system. Now, this particular system is again qualitatively, the response curve is going to look like this itself, however the times at which we get a curve of this sort is going to be 0, 1000, 2000 and so on and this is going to be 5000.

So, this particular system responds very quickly, this particular system responds very slowly. Now, let us say, that we were to solve this by Euler's method for solving this by Euler's explicit method. In this particular case, we would have to take h of the order of say 10 to the power minus 3 or 10 to the power minus 4 in order to get stable solution and in order to get stable solution with a fair amount of accuracy.

(Refer Slide Time: 39:52)

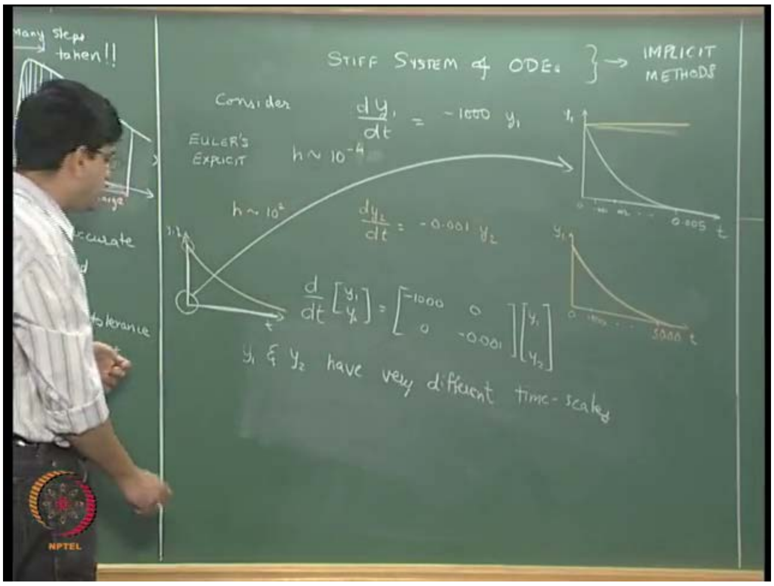


Whereas in this particular case, if we take the step size of 10 to the power minus 4, we need to run this particular method up to time 5000 as a result the number of steps with h equal to 10 to the power minus 4 for this particular system is going to be equal to approximately equal to 5 into 10 to the power minus 7.

Now, we do not want to take, so many steps for this method, as a result the h for the system we will take of the order of 10 to the power minus 1. So, this if this was only one equation that we had to solve we will take this h equal to 10 to the power minus 4, if this was the only one equation that we want wanted to solve, sorry not h to the power minus one it should be h to the power 3 minus 1 which is a h to the power 2, that is the sorry 10 to the power 2. So, that is the h that we will use for this particular system. So, if this system and this system were to be solved independently, we did not have any issues, we can choose two different values of h, we can solve them independently and that is the end of it. Now, however, if we had an equation of the form d by dt y 1 y 2 equal to minus 1000 00 minus 0.001 y 1 y 2. So, what happens over here is that y 1 and y 2 have very different time-scales.

So, y 1 and y 2 evolve at very different time-scales compared to each other y 1 is evolving in about 5 milliseconds whereas, y 2 is evolving over a period of an hour. Now the question is what is the value of h? We are going to take if we take the value of h equal to 10 to the power minus 4, we will end up taking the we will end up requiring almost 5 million steps in order to reach the solution at time t equal to 5000, if we take h of the order of 10 to the power 2, this particular method is going to go unstable.

(Refer Slide Time: 43:07)



So, because of this stiff system of o d e’s the definition of the stiff system essentially, is the stiff system is one in which there are components, which have very widely varying time-scales of evolution. So, if you have a multi-variable system, you can take, you can differentiate the system and get this particular guy over here find the Eigen values of this particular matrix the ration between the absolute value of the highest Eigen value divided by the absolute value of the lowest Eigen values is going to give you, how much stiff this particular system is if this Eigen value divided by this Eigen value is of the order of 10 to the power 4 or 5 or higher, in that particular case the system is going to be fairly stiff and we will need to use stiff solvers.

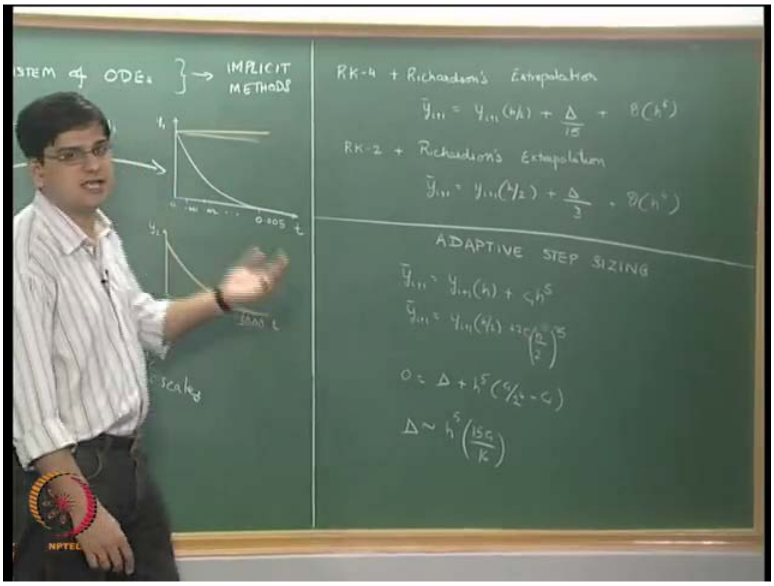
So, for this particular system the evolution, the way it is going to be happen is like, this if you were to plot y 1 y 2 versus t y 1 f t, we look at up to 5000 steps, what is going to happen is that y 2 will evolve in this particular manner, whereas, y 1 is going to look like this, it is going to look like it is a vertical line and a horizontal line, if we zoom into this particular region, if we zoom into this region. This is what we are going to get at y 1 does evolve but, it evolves in 0.005 times and not in 5000 times and in this small zoomed out period if were to look at this particular part in the zoomed out period, what we are going to see is that y 2 is more or less constant if not even going to have the slope it is going to be more or less constant in this particular time-scale. Now, these types of problems create a fair number of numerical difficulties and that is because, we need a small step size for stability of this method whereas, we need a large step size. So, that we have the overall solution reaching the steady state fast enough.

Now, Euler's explicit method or any of the explicit method is going to have a problem use in stiff systems, because it is going to require very small steps as govern by this the fastest mode of the system. So, what do we do is the solution for stiff o d e's is to use implicit methods. So, that we do not have to worry about the stability of these methods at all. We only have to worry about the accuracy of the system. In order to get more accurate solution in this initial region, we can have through the adaptive step sizing, we can have a lot of steps taken in this particular region and in the rest of the region, we can actually take much larger steps. So, that both the evolution of y 1 as well as evolution of y 2 we will be able to predict fairly well.

Now, the problem is why we require an implicit method and not an explicit method is even, if we take with an explicit method, if we use adaptive step sizing over here the solution would not have gone to zero, in the finite amount of time. So, when we take start taking larger step size, what is going to happen is that the solution is going to be unstable.

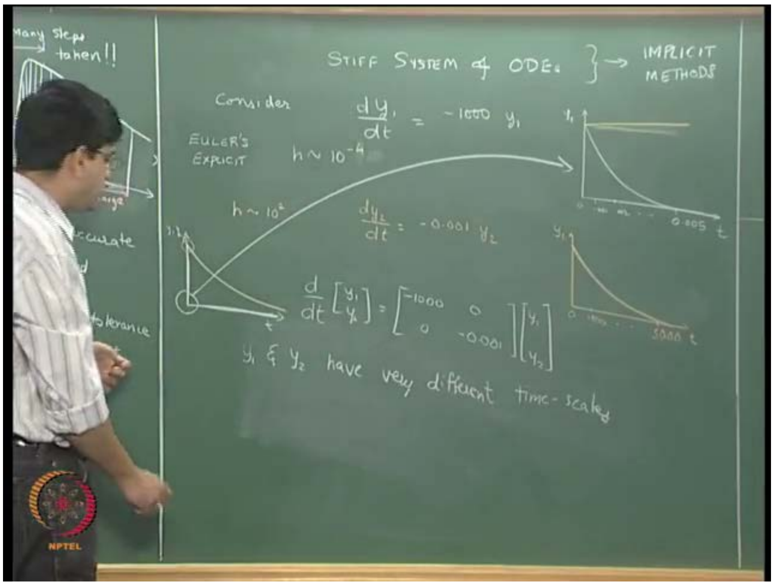
The reason why we cannot use explicit method is for h greater than t 10 to the power minus 3 or 2 into 10 to the power minus 3, this equation is going to go unstable. So, we cannot use a larger step size because as governed by this equation and if we take a smaller step size as governed by this particular equation, we are going to take a very large number of steps in order to reach the final value.

(Refer Slide Time: 53:31)



So, this is essentially, that all I wanted to talk about the stiff system, of o d e's finish of the lecture over here, what we have done here, we started off with talking about the global and local truncation error. Solve, the where the geneses of global and local truncation error comes about then we talked about the Richardson's extrapolation method in order to get more accurate solution. The Richardson's extrapolation method depended, on difference between the error using, a more accurate method and a less accurate method that difference then we said can be used in this adaptive step sizing type of an idea, where we change the step size based on the difference between y i plus 1 computed with h by 2, as the step size and y i plus 1 computed with h as the step size, that is what we do in adaptive step sizing.

(Refer Slide Time: 53:48)



And finally, we talked about stiff system of o d e’s and stiff system of o d e’s are the systems, where we have multiple time-scales, accruing in the same system. The systems might be linear or non-linear. Nonlinearity typically, makes the problem even more worse, than in the linear system, but stiffness is independent of linearity or nonlinearity, it is independent of the actual values of y 1 and y 2, it is depends only on what are the time-scales at which y 1 and y 2 evolve.

And finally, we said that the explicit methods are going to be not very useful, when we are going to use, when we are trying to solve stiff system of o d e’s and we have to use implicit methods, so that is where I end this lecture today.

Thank you.