SCIENTIFIC COMPUTING USING PYTHON

Professor Vivek Kumar Aggarwal & Professor Mani Mehra

Department of Mathematics

Indian Institute of Technology, Delhi

Lecture No. 35

Welcome, everybody, to Scientific Computing in Python. So, like in our previous lecture, we discussed the Euler method, the modified Euler method, and the Runge-Kutta method. Today, we will go beyond that and expand it to a system of equations. So let's get started. So in the lectures that we had till now, we have taken only a single differential equation. But what will happen? If we have this type of equation to solve, let me take an example. Today we will do system of equation. then what is it? How to do this? Now as you might have seen, suppose I have this equation a second order initial value problem, someone I have the value y' - 3y' + 2y = 0 and here the condition is initial -1 and y' at 0 is given to me that is 0, suppose this is given. Now we have to solve this. Ok? Since this is a linear equation, it will be solved very easily. But now we have to solve this numerically. So till now the above methods we have done are for first order OD. This is a second order OD. So what do we do now to solve this ? We have discussed this like this before as well. We will now convert this to the system of order. How about first order audio ? So we always do as we assume here that y is suppose some y1 and the dash of y_1' is suppose y_2. Okay, so let's write it like this. Now let's write what I wrote y1, I wrote y. Now let's see what y_2' will be? Now look, from here, if I want to take y_2' then what will that be? the y_1'' means y_1 was y y'' okay? So from here, if we want to write y_2, what does y'' and y'' become? If we take this to the right side it will come to $3y_2 - 2y_1$, okay? and what does y_0 mean? What is $y_1(0)$ and y_1'? y_2 will come from here $y_2(0)$ which is zero. So now we have a system of OD [y_1 y_2] here the derivative here is my matrix. A [0 1 -2 3]is a vector whose values are y1 and y2 this is our solution and we have y1(0) y2 (0) and the values of this vector are this,

System of ODE

2nd order IVP $\begin{cases} y'' - 3y' + 2y = 0 & y(0) = -1 \\ & y'(0) = 0 \end{cases}$

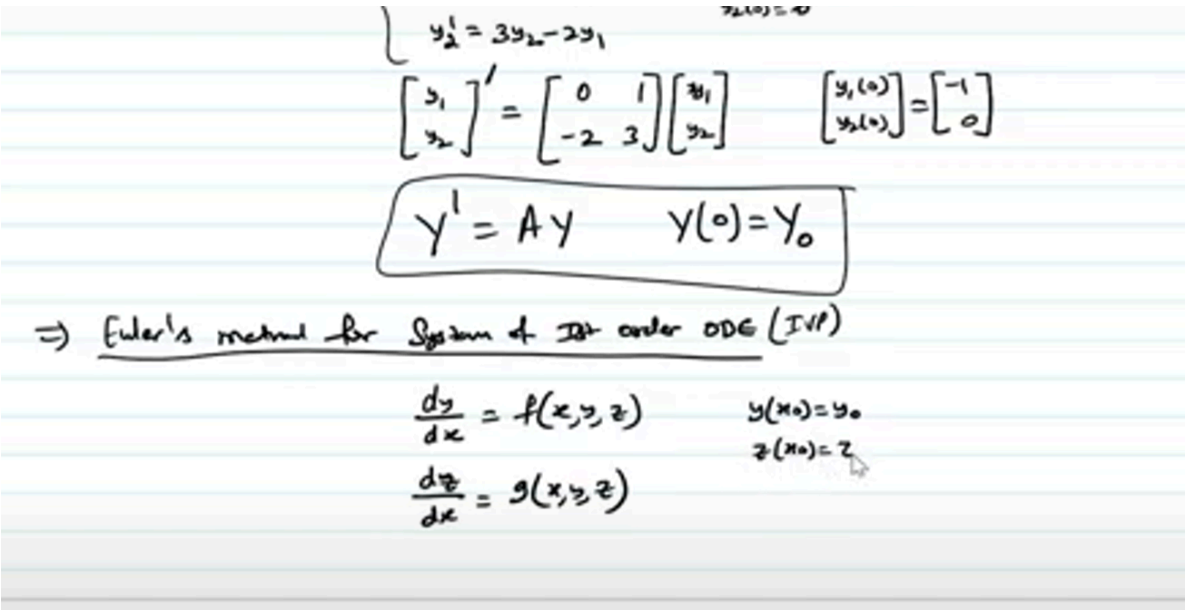
assume $y = y_1$, $y_1' = y_2 \Rightarrow y_2' = y_1'' = y''$

$y_1' = y_2$ $\qquad y_1(0) = -1$

$y_2' = 3y_2 - 2y_1$ $\qquad y_2(0) = 0$

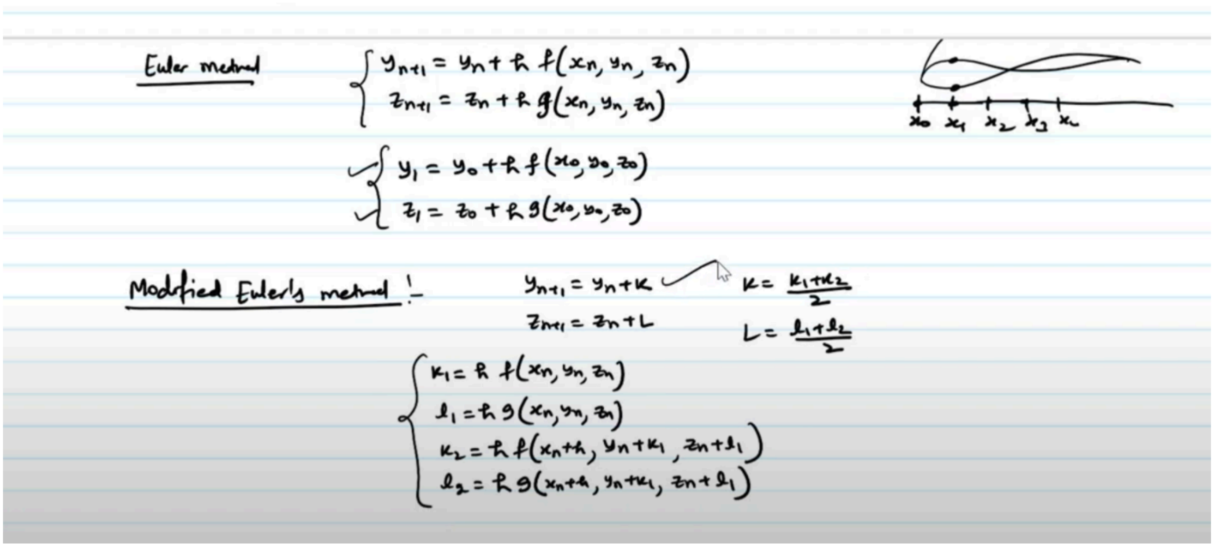
$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}' = \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \qquad \begin{bmatrix} y_1(0) \\ y_2(0) \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$$

okay so now if we see then we have this y'=Ay, okay because in this the right hand side was zero so it is homogeneous and y at 0 is y, so this y is a vector which in this case is a 2 / 2 vector like a matrix. So 2 cross 1 is a vector. Like this we will get third degree punishment. So 3 y1 y2 becomes y3. Of 4 degree, of 5degree. Thus, we can take any initial value problem of any higher derived order. We can always convert it into a single system of single ODEs. And now after that we have this. Now look, this is single ODEs. So what do we do with it now ? We can solve this only by applying Euler's method. We can solve this by applying modified Euler method. We can do this using the runge Kutta method. So now we have to do the System of Odds for what it is. Ok? So now let's suppose I do this. So euler methods for systems of first order, now we have to solve the systems of first order OD. There is ODE and initial problem. The condition is given. Only then will our unique session come. So this work we are doing for two, we can do it for three also, we can do it for four also. So suppose we have what we have now? y1 and y2. So just like that I assume that we have a differential equation. I write it like this. Let me write for 2/2. So suppose we have dy / dx = some function of x y and z and this is dz / dx and here we have some function here we are given the condition that yz such that the condition is given zx x0 is z



Okay, so here we have a system which has two equations. There are two odes. Ok? And there are two variables s, we can also have a third y, z, w, so in this the independent variable we have is one, but the dependent variables x, y and z become two, so we also have the initial condition. Similarly, if you observe carefully, we have a System of OD. For example, if you work in mathematical modeling, you will find many models in it, such as SIR models and population models. All the mathematical models that we use in real life, they are all in the form of first order ODs. Ok? So we can write it in this form only. And these functions that we have, f and g, can be linear or non-linear. So the methods to solve this are analytically known to us, which we can do using matrix methods, their eigen values etc. But what we have to do in this case is to apply the oiler method. So you see what would be the Oiler method ? Now we have to calculate y(n + 1). So y(n + h) f will now replace x with xn. yn will replace y and zn will replace z. Just like this we will have z(n + 1), this will come to us zn +, okay? Now this is our quantity h, now this will come g, this will come xn yn and zn, okay so we have got this value. So what do we have? The independent variable is x. So in that we have x0, x1, x2, x3, our mesh points will be calculated like this. And what next? Something with respect to a solution y, this is the solution. This is some session. Second, there is another solution of y with respect to y z. In this way we will get number of solutions. Now depending upon what our system is like? So now I had taken 2 to 2 system, there we made this using oiler method. Ok? So now look what will happen to us as y1? y1 will be y0 + h which is what we will calculate. f Now we know x0, we know y0, and we know zero. So we will substitute and from there we will get the value of y1. Now I have to calculate z1. So what do I do to calculate z1 ? z0 We know this. h will remain the same because it is behaving on x. g will come now and what will happen in g? We know x0, we know y0 and we also know z. Ok? So what did we do? From here we have calculated the values of y1 and z1. Suppose this is y1 and this is y and this is z. So suppose these values came from here. Now using these we will go to the next one. So now we have to do the calculation twice in every iteration. One for y1 and one for z1. Ok? So this will become our Euler method. Like this we can create modified Oiler method. Modified Euler Method. Now in the modified Euler method, what will we do first? yn+1 what happened? We have to calculate yn + k. and what is zn + 1 to be calculated?

what happened? I write this as zn + K. I write L instead of K. Now what is K? K is the average of k1 + k2. What is L? Average of l1 + l2. Now it depends on how you will find out these values. So we have k1, we know that it's going to come out at h f(xn,yn,zn) and l1 what is that? h g(xn,yn,zn) will come out on zn. Now let's see what k2 is? Hf Now look what's here? Xn+h is ok? So basically it is xn + 1, so the value will be obtained on that. So here this yn + K and Zn + L will come. What will be l2? h of g now here it comes xn+h means next one yn + K and zn + L this is okay. So now we have to calculate this. So we will calculate this and then substitute it and we will get the values from here. So similarly we have two function, three function, four function, if we have a system of OD, like the SIR model, it has three equations. So three variables will come. Ok? So there will be three such variables.



[13:11]

So, what this means is that if we have to calculate this then 6 (k1 k2 K3) will also come. l1 l2 l3 will come. So, this means k1, k2, k3, the values are f, g and after that another function will come. This is the calculation that will have to be done in this modified euler. Ok? So we can calculate it like this. Similarly, we can also define the Runge Kutta method. So how do we calculate in range put methods, in the same way whatever values we have, like we have yn + 1 then what will come out to be yn + 1 / 6, this was k1 + 2k2 + 2k3 + k4, zn + 1 will come out to be zn + 1 / 6, here I have written l1 + 2l2 + 2l3 and l4 will come out to be this. Ok? So we will calculate this. Now to calculate what is K1 ? That would be h of f (xn yn zn) , l1 what would be? That will be hg (xn yn zn), k2 h of f now look this will be xn + h/2 yn + k1/ and zn + l1/2. what will be l2? will be l2. h of g same as xn + h/2 yn + k1 / 2 and zn1 + l1 / 2 or is it correct? So if you give it to us carefully, then this is what we calculated. Ok? So alpha1 was half and beta1 was half. So basically this is happening. So now I am writing it directly. So here it comes, what will be k2 k3? Even in that, if you pay attention, it will also appear as half. So this will come xn + h/2 yn + k2/2 and zn + l2 / 2, this will come here l3 will come h of g and these are the arguments, they will remain same here, okay so this xn + h/2 yn + k and zn + L/2, this is k4 , h f now here this quantity one will come so this will become xn + h and this becomes yn+ k3 and zn + l3 and here we have found out L4 and h of g so xn + h yn + k3 and zn + l3, this is it. Ok?

Runge Kutta method

$$\begin{cases} y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ z_{n+1} = z_n + \frac{1}{6}(l_1 + 2l_2 + 2l_3 + l_4) \\ k_1 = h\, f(x_n, y_n, z_n) \qquad l_1 = h\, g(x_n, y_n, z_n) \\ k_2 = h\, f\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}, z_n + \frac{l_1}{2}\right) \qquad l_2 = h\, g\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}, z_n + \frac{l_1}{2}\right) \\ k_3 = h\, f\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}, z_n + \frac{l_2}{2}\right) \qquad l_3 = h\, g\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}, z_n + \frac{l_2}{2}\right) \\ k_4 = h\, f(x_n + h, y_n + k_3, z_n + l_3) \qquad l_4 = h\, g(x_n + h, y_n + k_3, z_n + l_3) \end{cases}$$

[17:43]

So we will calculate it from here like this. We will substitute here and we will have yn + 1 and zn + 1. So now we can apply the Runge Kutta method which is of fourth order. So, we can calculate the system of equations. So now let us see how we are calculating this with its help. This is how I take supposed examples. Let me take this one as an example. Of the third order we have that which is ODE. One, this comes to us and we have the condition given that y 0 is 1 y' 0 2 this is the initial condition given y' 0 is 3 this is given suppose so now we have to solve this so now see what we will do, I will take y as y1 and I will take y1 as y2. I will take y2 to be y3. So now see what comes here? y1 d is y2 y2' is y3 What did y3 become? will become the triple derivative. Ok? So this basically becomes x^2 y1 - 4y2 + 2xy3 and this is done and the + 1 is coming here. Ok? And what will be the initial condition here ? y1 0 is 1, y2 0 is 2, and y3 0 is 3. So here we now have a system with three unknowns. So if we pay attention to this then I can write it like this. y1 y2 y3 d this becomes the matrix. This will come as [0 1 0, 0 0 1] and here we will get x^2 - 4 and this 2x will come as y1 y2 y3 + 0 0 one so it has become nonhomogeneous. Ok? So this is a system of non- homogeneous first order audio and this initial condition is reached. So now we will solve this. Like we used the same method for two equations. So what will happen in this? The calculation will increase. So this is how we will find it out. I will calculate it.

Exp $\quad y''' - 2xy'' + 4y' - x^2 y = 1 \qquad y(0)=1,\ y'(0)=2,\ y''(0)=3$

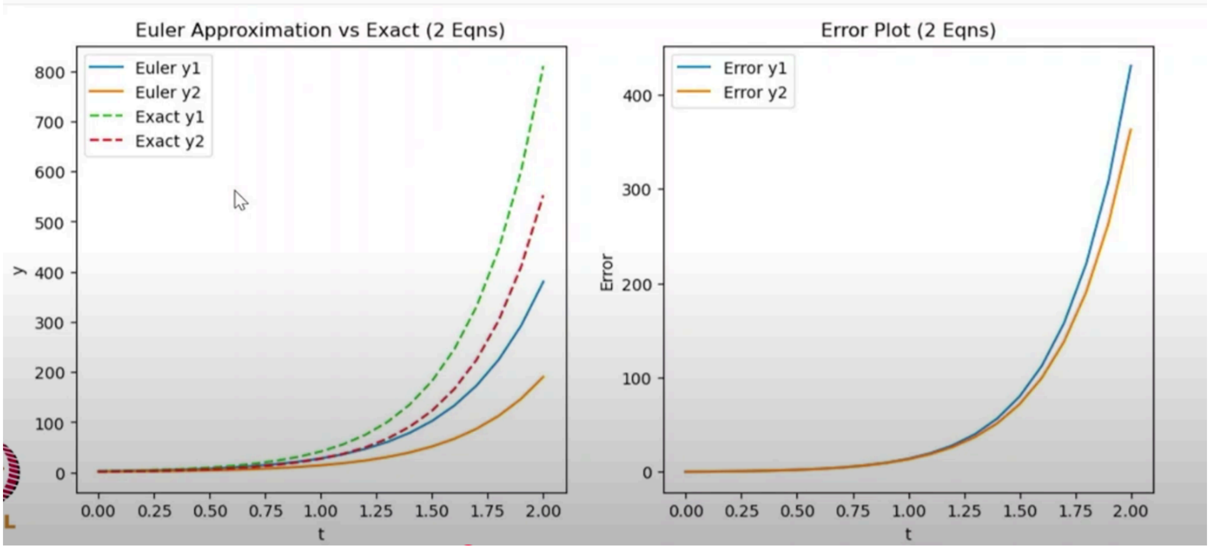
$y = y_1 \qquad y_1' = y_2 \qquad y_2' = y_3$

$$\begin{cases} y_1' = y_2 & y_1(0)=1 \\ y_2' = y_3 & y_2(0)=2 \\ y_3'' = x^2 y_1 - 4y_2 + 2x y_3 + 1 & y_3(0)=3 \end{cases}$$

$$\Rightarrow \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}' = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ x^2 & -4 & 2x \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

[20:41]

So now let us do this with the help of code and see what will happen if we calculate this with the help of code ? So this is like solving the system of OD and using Euler's method. Ok? So now we have the Euler method but we have written OD with capital Y. Ok? And the function on the right side is ft and y and capital y, we have already found out that they can be like y1, y2, y3. After that we will define all the parameters given to us. Ok ? and what is in the result? We will return the values of t and the values of y because that is our approximation. So all these will be the same. What do we have now? There will be two values within y0. Ok? So the values of y1 and y2 are, now see, we have calculated this, if we apply Euler method then it comes out to be y values + h times this. Ok? So we have accepted this function as f. So the first component of f is f and the second component is g. What we have just written is an example. Ok. So we return to it the values of t and the values of y. Now see, we are defining equations. So dy / dt we define as y 1 + 4 y2 and here we go. Ok? The one we have defined for the second y1 + y2, it is calculated like this and will return d/ dt which is our system, okay, we have defined the right hand system, after that we have defined the values. t0 Whatever it is, I took it as zero. We have defined y0 as the initial condition two and one. The value of h is t and the value of h is two and I took the value of h as one. And we called oiler methods from here. Now it depends whether we have the exact solution or not. So if the exact solution is non, then we have defined it here. After that, we calculated all the values that would come to us and made a plot of it. So okay what do we have in the plot? I will write all the plots. 1 2 1 The first one is y1 y2 The second one is showing the error. Ok? So like this, we have defined its third face portrait also. Face portrait is that the relation between y1 and y2, the relation between them is called face portrait. Now we have taken the second system of three equations, so we have taken this. Ok? So what's in it? I took y1, y2 and - y0 - y1. So this is how we will calculate the third equation and we will call it what it is. So we solved it. Lets see. So this is the solution we have. So what is the solution of those two equations? Oiler y1 this one and this y2 this is the computed solution that we have. And these dash dash dash lines are the exact solution that we have. And their error plot is this.



[24:35]

Ok. And after we created the face portrait, this is what came out. This means that there is a linear relationship between the two. Ok? So we can calculate like this. Now I can change this. Now as I defined the examples. So with the help of that example, let us see what we have to do now. So now let me do the first calculation. So first of all we will see, I am defining the system that we have, I will take it as y[0] + 4 times y1, okay so I will take this as mine, I will multiply it by 0, okay and this which has come is one because if we look at our system, then this was coming. Now I am trying to solve this one. So now see what is y1d? y2, so y2 means its coefficient, the coefficient of y1 will be zero and the coefficient of y2 will be one. Only then will we be able to write. Similarly y2 d then what will happen in it? -2 and 3 should come. So let's see what will come, so here -2 and three will come, after calculating this, so the system which we have formed, is formed, okay, so depending upon what system we are taking, now what is our initial condition, t0, so let's say it is zero, it will always be there, y0 which we have taken, we had taken it, I will define it here -1 and this 0 which was our initial condition, okay, so we will have to define it, the initial condition -1 and 0, so I have defined this and tn will be taken only till two, there is no problem in it and h which we have taken, we have taken one, okay, so now the exact solution of error is that we have not written, so if the exact solution is non, then we write it here and if it is unknown, then we do not write it.

```
    return dydt

t0, Y0, t_end, h = 0, np.array([-1, 0]), 2, 0.1
t_vals, Y_vals = euler_method(system1, t0, Y0, t_end, h)

# Exact solution for comparison
def exact_solution1(t):
   # return np.array([np.cos(t), -np.sin(t)])
    #return np.array([2*np.exp(3*t)+(np.exp(t)-np.exp(-t))/2, -np.exp(-t)/8-np.exp(t)/4+(11/8)*np.exp(3*t)])
    return np.array([np.exp(2*t)-2*np.exp(t),2*np.exp(2*t)-2*np.exp(t) ])
exact_vals = np.array([exact_solution1(t) for t in t_vals])
errors = np.abs(Y_vals - exact_vals)

# Plot results
plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
plt.plot(t_vals, Y_vals[:, 0], label='Euler y1')
plt.plot(t_vals, Y_vals[:, 1], label='Euler y2')
plt.plot(t_vals, exact_vals[:, 0], '--', label='Exact y1')
plt.plot(t_vals, exact_vals[:, 1], '--', label='Exact y2')
plt.xlabel('t')
plt.ylabel('y')
plt.legend()
plt.title('Euler Approximation vs Exact (2 Eqns)')

plt.subplot(1, 2, 2)
plt.plot(t_vals, errors[:, 0], label='Error y1')
```

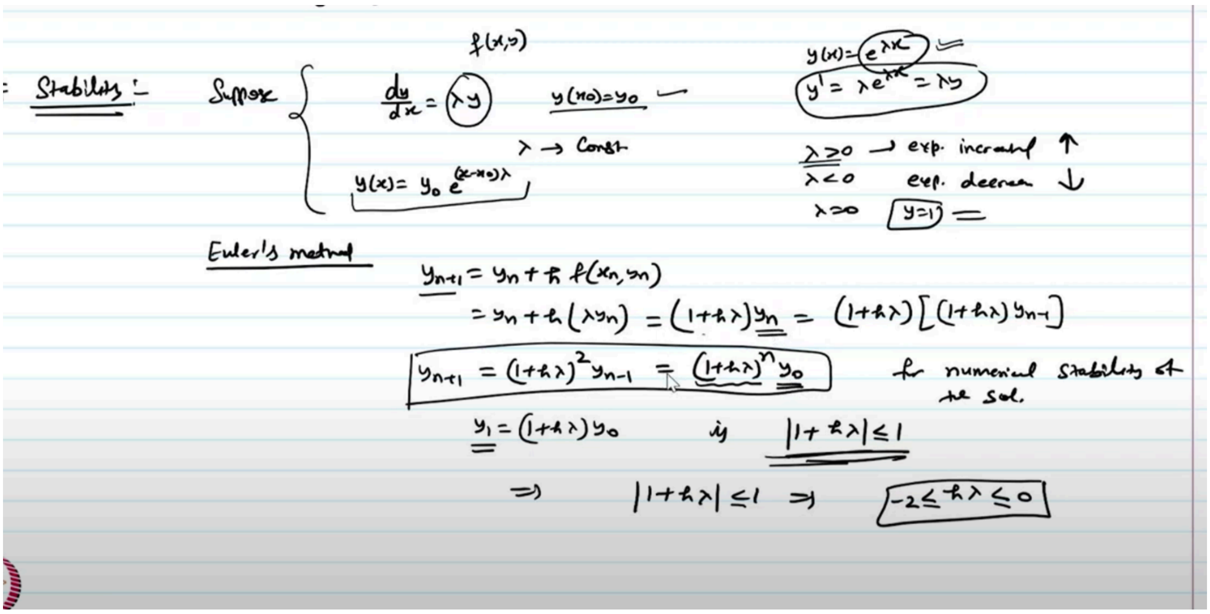
[30:13]

Ok, so with its help we can calculate it. So if I look at the values in this array, it looks like e to the power of 2x - 2 e to the power x. Let me try once to define it, so return np dot array, in that the first value that I am taking, I am taking np dot exponential to star t, this is coming, okay, so this is the first value and the second minus 2 * np exponential and this is what we got, so this is the first one and the second one is zero, the second one will actually be its derivative. It will not be zero. must be its derivative. So I'll take this one. Control C Control V because we had assumed its derivative will come here two times of this, no it is not here, this is okay, so NP two times of this and this will remain the same, so this will come because y2 is just the dash of y1. So y1 is this np. Exponential means e to the power of 2t - 2 * e to the power of t and if we take its derivative then the multiplication will happen beyond two and this one will remain the same. OK so it came to us. So here we have the exact session, let's run it. So we solved it and see this is coming. So we got this error and this solution came. So initially they are starting well and after that errors are developing. That is what error develops, we have seen it here. So we got this solution. Is it okay? So in the same way, we can do this third equation as well. So, we can calculate this three system, the third equation. So I am calculating this. So this we have NP okay. I took this. I took this. And I have defined it. So what do we do now ? So we will continue calculating it in this manner. y1 y0 y1 y2 Now if I want to define it then I can define it like this. So this is y2. This is y3 and the dash of y3 which we had calculated will come out in terms of this. Ok? Ok? So in this case the quantities that we have are y0, y1, y2. So once I try to apply this system. So this is with us and I will take control of it. Control V and I will comment them all out so that they don't get used. Now see what is this? Now this is okay. y1 yt Now we have to see the value here. So here we have t^2 multiplied by y0. Minus 4 times it was. Plus 2x2 * t* which is y, that is the value that was coming out. Is it okay? And this was +1 because it was a +1 value. It had arrived. So see from here, we have defined it. So how do we write t^2 y1 - 4y2 ? It's written from here . This one. So I am trying to solve this quantity. are you ok now? So we have calculated y1, y2 and y3 and these values 1, 2, 3, we give these values here. So this value will come here 1 2 3 and the y end, suppose I give one and h, I have taken one. So these values are given to us. Ok? Now let's see how to solve this. So this came to us and the third solution that is coming for oiler is this. Ok? So we do not know its exact session. So we calculated this. So this is the values. Why? Because you will see it was starting from one. This is starting from two and this is starting from three. So the initial condition is fine from here. So this is what Oiler Y1 is, so it's starting from one. This is what Y2 is. It's starting with two. And the green one is Y3, it starts from three. And if we look at the behavior of all three, this solution is coming. Ok? So, these value solutions have come to us. So like this we can calculate it with its help. You can also find out its exact solution. And he can write. And I try to take this if I end up at 5 instead of one. And I'll increase the H a little bit. So let's see what is calculation. So here is the solution. So the solution has become quite extended. And from here onwards this DK happened. Ok? So these values are quite good, this has become 10 power six, so it means it has become quite small, it is in negative, right? So this value starts from zero, after that it decreases, so these are our solutions, so we can play with it and calculate it, so in the same way we can apply Rangokatta methods, so we will calculate Rangokatta method for single equations, so what is there in the single equation? The values of t will come, the values of y will come. And look, this is our main purpose, this is this. So what have we done in this? k1, k2, k3, k4 all have been defined. After that, see what will be the values of y? y + k1 + 2 times k2 + 2 times k3 + k4 / 6 and this is where we increased the value of t. So we have the values of y and the values of t, which are the updated values, we will return them from here. So we have calculated this.

```
    y_vals : array - Approximated solution at each time step
    """

    t_vals = np.zeros(n+1)
    y_vals = np.zeros(n+1)
    t_vals[0], y_vals[0] = t0, y0

    for i in range(n):
        t, y = t_vals[i], y_vals[i]
        k1 = h * f(t, y)
        k2 = h * f(t + h/2, y + k1/2)
        k3 = h * f(t + h/2, y + k2/2)
        k4 = h * f(t + h, y + k3)
        y_vals[i+1] = y + (k1 + 2*k2 + 2*k3 + k4) / 6
        t_vals[i+1] = t + h

    return t_vals, y_vals

# Example ODE: dy/dt = -2y, y(0) = 1
def example_ode(t, y):
    return -2 * y

# Example ODE without exact solution: dy/dt = y^2 - t
def complex_ode(t, y):
    return y**2 - t

def solve_and_plot(f, t0, y0, h, n, exact_sol=None, title="Runge-Kutta 4th Order Method"):
    t_vals, y_vals = rk4(f, t0, y0, h, n)
```

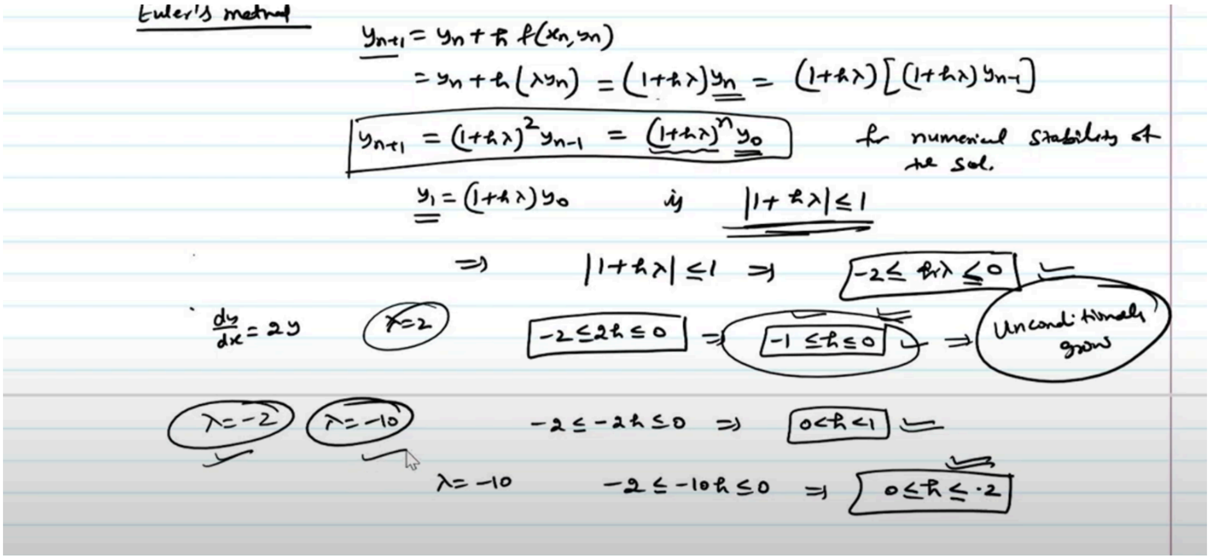
[35:56]

Ok? So we are solving ODE. Suppose dy / dt = -2y this is what we are doing. And its exact solution is also known. So we applied for it and after that it became ours. So these two are matching exactly, they are overlapping. And this error is very small, 10 to the power-10, here it is going up to 5 * 10 to the power-10. So the error is improved. Ok? The exact solution without the exact solution with the exact solution. So we can find out the error and otherwise we can calculate it. So what will happen in this case is that we can have exact solutions which are known as well as unknown. Depending upon us, we can define what it is. So this single is for our quantity. Similarly we can define it for the system of ODE. Ok? So, we can always define these things very easily for the higher dimensions. Now we have to pay a little attention. Sometimes we have seen that sometimes I am taking point one, sometimes I am taking point two. So how do we know what we need to take? So that our solution should always exist. Ok? So for that, as we discussed earlier, actually three things are always involved in it. Consistency, Convergence and Stability. So, if two of these things are already available to us, then the third one is available automatically with the help of Max Equivalence Theorem. Ok? So now the main thing we need to do here is we want to see how do we define h ? what should I take? Sometimes I'm taking one, sometimes I'm taking 2. So can we take H as 5 also? Can I also take H Coro1? Ok? So how do we see this H? So for this we have a criteria and that is stability. Ok? So we have already defined stability that if we have a method that we will call it stable only then the entire error, whatever the cumulative error is, should be bounded and not unbounded. Because you have seen that y0 is being used to find y1. Then y1 is being used in y2. Then y y2 is being used in y3. So if we have a little error in the beginning, then we say that that error should remain bounded. It is not like that he will become unbound. So if it becomes unbounded then it means the method is not stable. Ok? So, we have to check this. So to check the stability, suppose I took a differential equation. Let us take dy / dx = y. And y at x0 is y0, I took it. Ok? And what is lambded is a constant. So you will know that if we solve this then we always get this solution. yx = y0 e to the power x - x 0 λ This will come out. Ok? Why? Because when I put x 0 in x then this quantity will become zero and then y will become 0. Otherwise this quantity will come. Because if we want to put a solution like y = e to the power of lambda x, suppose I 'm putting this, then what will be y'? That's going to be lambda e to the power lambda x. So what does that become? MBDA became y. So this means that this solution will definitely be like this. e to the power λx but we have to define the initial condition that it is satisfied, so our solution should start from x0, so its solution will come in this form, so we can very easily find it because it is a linear equation, a simple equation, now it depends on what our solution will be, if I take lambda 0, then what will it mean, our solution will become a constant, if I take lambda positive, then we will get an exponentially increasing study solution. So depending upon whether our lambda is positive. Lambda is negative. This is exponentially increasing, this is exponentially decreasing, so this will be dk. This is okay? So lambda = 0 then the quantity there will become a constant. Ok ? So y is going to become one. Depending upon what is our initial condition? So in this way, we have to see how the solution will behave for the different values of lamda. Now if we look at this case, we know that if we go a little into the background of this equation, then lambda is positive. Ok? So what happens in this case ? The stability of our differential equation will become unstable as t goes to and x goes to infinity. It will go to infinity. And if h, which is our lambda, is negative then it will be dk. So whatever solution we have, it will be stable. Ok ? So we already know this thing. So now let us find this out with the help of Let's Euler method. So now we apply the Euler method. So when we applied Euler's method, we got yn + 1 = yn + h f (xny), this is what we get. Now this is ours. This is our f. So I will calculate this. So see what comes? What will we get in place of yn + h f ? lamda yn has arrived. Now I took yn as common so I got 1 + h yn this came to us. So this means that if we multiply the solution which was at the nth step by this, then the solution will come at the next step. Now I can also write it like this 1 + hmλ^2 yn - 1 why? Because I can write it like this. One, these are constant quantities. And instead of yn I can write 1 + hλ yn - 1 then it will be square. So after all if you see n then this was one. If n is - 1 then it becomes square. So if I raise this to the power n of 1 + h leλ then it will become y0. Ok? Well if I repeat its value again and again then it will become quantity. Now you see, we know that our initial solution was non. So what this means is that y1 will become 1 + λ * y0 when multiplied by it. Ok? So now we want this quantity to be yn + 1. So if we want that our solution should not grow too fast, that is, it should remain bounded and should not grow unboundedly, then this is possible only when our quantities are within bounds. If these quantities are within bounds, then we will definitely know that our solution will not become too big at any step. will not become unbounded. So if we look at this and I assume that if 1 + h lambda, its quantity, if it is less than one, then it can be negative as well depending upon what is lambda, so if we take its modulus, then if this value becomes less than one, then if we take its power then it will become smaller, so if the solution that we have is y0, then the next solution will be smaller than this and will not grow too much, okay? So in the same way, whatever solution we have, it will not be unbounded towards what it is infinitely. Ok? So if we want this, we need stability in the beginning, so we should have this quantity. So we can write that for the numerical stability of the solution we need to have this quantity. So from here we have that 1 + h lambda which is less must be < 1. And from here, if you look, I here will become hm λ - 2 and this will become 0. So this quantity has come, you can take equal to it also. No problem. What's too much? It will become equal to one. So the same values will keep coming. y0 y1 y2 all the beans will come.



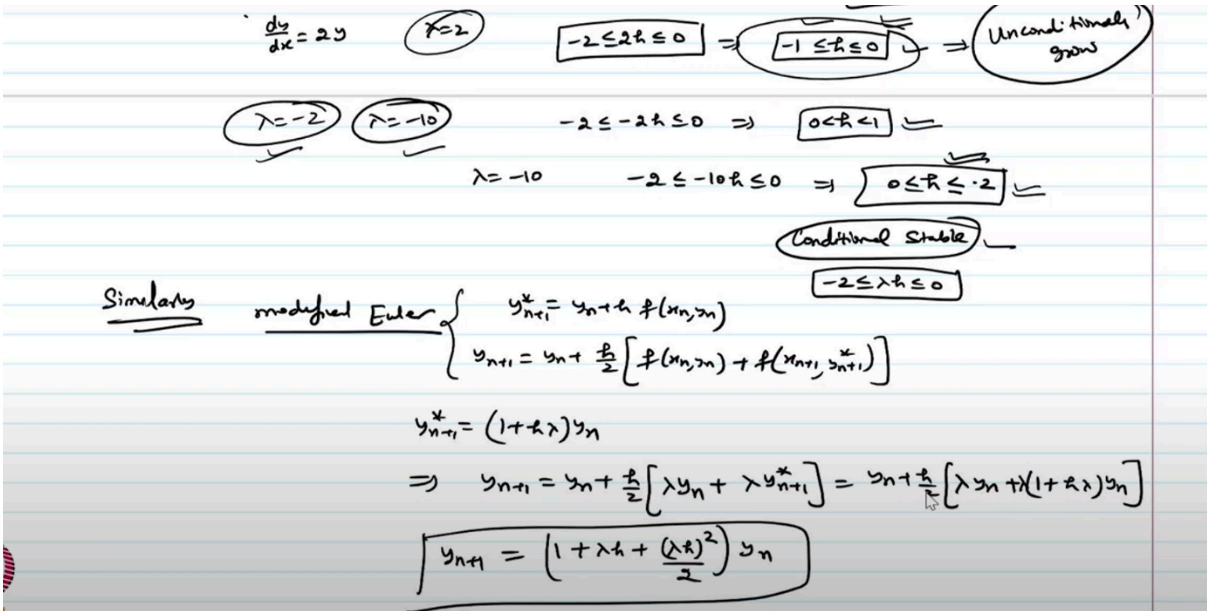
[47:03]

But they will not be unbound. So we can take equity also. So this is the quantity that we have. Now if we look on the basis of this quantity, now we know that h. h we have to find out. Suppose lambda will always be known to us. So if we look from here, what has happened to what we have ? -2 /λ 0 /λ This is the quantity when the lamed we have is positive. So that is why we never write it in this form because we keep both the quantities in mind that lambda can be positive as well as negative. So this region, this region has come. h, this is the region which we have, this region which we have, this stability region we have for the Euler method, so now depending upon what value of lambda should we take? Now suppose lambda is the equation that I have, suppose I solved the differential equation dy / dx = 2y so the lambda came out to be 2. So here the quantity comes from this, we should have 2h - 2 and 0. So, we will define it accordingly to our h. So whatever our h will come from here, it will come from -1 to 0, so we can define h accordingly. Ok. So now this is ours. Now everything depends on how we define our values. What happens if h is lambda, which is -2 ? Ok? What happens if lambda is -10 ? Then you will see the value of h, now in this case, you will see that if h lambda is positive as I told you, then in that case it will always grow, so in this, if h lambda is positive, then it will always grow up unconditionally, it has to grow because it is an exponential increasing function. But depending upon what is the lambda if the lambda is negative. So if suppose lambda is negative. Suppose lambda is -2. So here we have -2 < = -2h < = 0 and here we go. So from here you see, the h that will come is between 0 to 1.



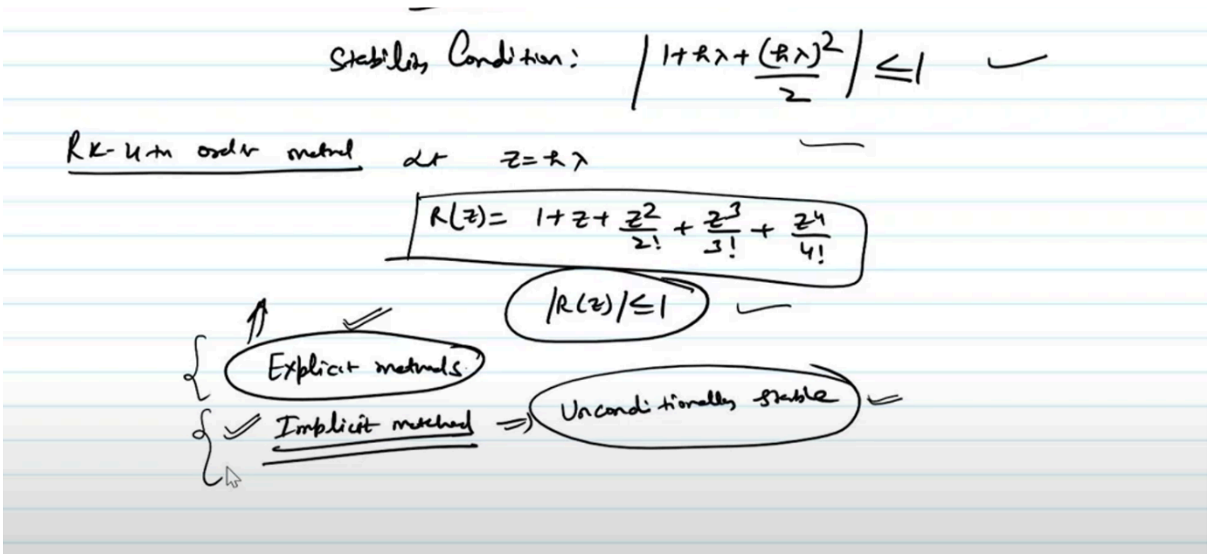
[50:30]

So if we keep h between 0 and 1 for the λ = -2 then we will always get our solution. Now if I reduce this thing further, I can write -10 instead of lambda. What will happen if I write -10 as lambda ? And if lambda = -10 then what will be the result? -2 so -10h 0 here it is. So now you will see that it will go from 0 to -2 here. So you see, now we will have to take the values of h in this. If lambda takes our -10, then now our h is what we have defined for the stability region. Ok? So from here we have to find h depending upon. So this means that our stability is conditional. So we call this a conditional table. What does conditional stable mean, depending upon the lambda, what do we have, so what is the conditional stability of the oiler, it has become -2 0, this is the stability of conditional oiler, so similarly we can see the modified oiler. What was in the modified euler? yn + 1 = yn + h / 2 f xn yn + f xn + 1 and yn + 1 star where yn + 1 star is yn + hf at xn yn it will come. Ok? So from here we know the value of yn star, so yn + 1 star has become 1 + h laλ yn. Ok? So now we will substitute it here. then what will be yn+1? yn + h / 2 f what will come is λ yn + laλ yn + 1 star and y is here so yn + h / 2 λyn + 1 + h laλ yn it has come. Ok? So now look here, if we calculate yn and h from here, what does this become? h leλ ok so from here it will become 1 + +λh whole square by 2 λ h yn this will come. Ok? So, we have calculated this. And lambda was also there here because this was there, so lambda * this, then lamλ was also there here, so now the calculation will be done from here, so lambda yn yn 2yn and lambda and h is also square from here, so this will come, so if you see, this quantity has come, so now in this case,



[54:45]

if we see, then we can go in the same way, so you will see that in this case, what has our stability condition become? 1 + hmλ + square of hmλ / 2 and here we have the condition. So what will happen with this? There was some improvement. Little by little we will see that the range that we have will increase a little. So it came to euler. Ok. So like this we will go to Runge Kutta after euler. So here I will simply write the Runge Kuta fourth order method. Whatever condition we get in that, what is it basically? Which is a quantity, late, because if we write it repeatedly then we represent it with z. h λ so if I write the stability region that we have in the form of z then it will come to 1 + z + z^2 / 2 z / 3 so in this case z 4 / 4 will come down to this point. And the stability region will be the equation of R(z). So, in this way, the stability regions that we have will keep on increasing. So, I will just show it to you with a graph plot that how can we do this, the stability region, this is the stability region of Euler's modified Euler, so depending upon what are our values, so after calculating this, this region of ours will come, so you will see that the one in red is between -2 to 0, so this scale means -1 was its center and a circle of radius one will come. So this comes to us. Next one was a modified Owler in blue colour. So see in this, the region has increased a little bit, our stability region and the R K H which is the fourth order rungee Kutta is green. So in green you will see that our stability region will be very large. So the larger the stability region, the more easily we can move. We can find out the values of h. So for higher H we will have stability. So all these methods that we have used till now, if we look at these methods carefully, then Euler modified Eulernge Kuta methods, all these methods are explicit methods. So explicit methods of solving the initial value problem always have stability conditions. So mostly these are stable but conditionally there are some such methods like implicit implicit methods, explicit implicit, so just like there are implicit functions, there are also implicit methods. So just like implicit methods, we take Euler and backward Euler as well. So those methods are unconditionally stable. It's just that we have to do a little more work in calculations competitively. But the advantage of the explicit method is that we can define and compute it very easily. The little bit of programming that is implicit in what happens unconditionally is a little complicated. Nonlinear equations have to be solved. But the advantage in this is that they are unconditionally stable. So whatever is h, we can take any value and do it. Ok? Ok, so depending upon whether we are using the explicit method or the implicit method. We can define its stability region



[58:53]

Ok? So we have seen that this method, if it is an explicit method, which is our Euler method, Runge methods, we have defined all these explicit methods, then we have defined its stability region, we have shown that if it will have the values of lambda h, only then we will say that our method is stable. If we define value outside that then there will be instability in it. But if we have an implicit method which we have not done, then what happens in it is that we apply the backward Euler method and with its help, our Euler method which is formed is implicit but unconditionally stable. So generally implicit methods are used for stiff problems and we will not do that in this course. So I hope you have understood today's lecture. You must have liked it and thanks for watching the lecture. Hello.