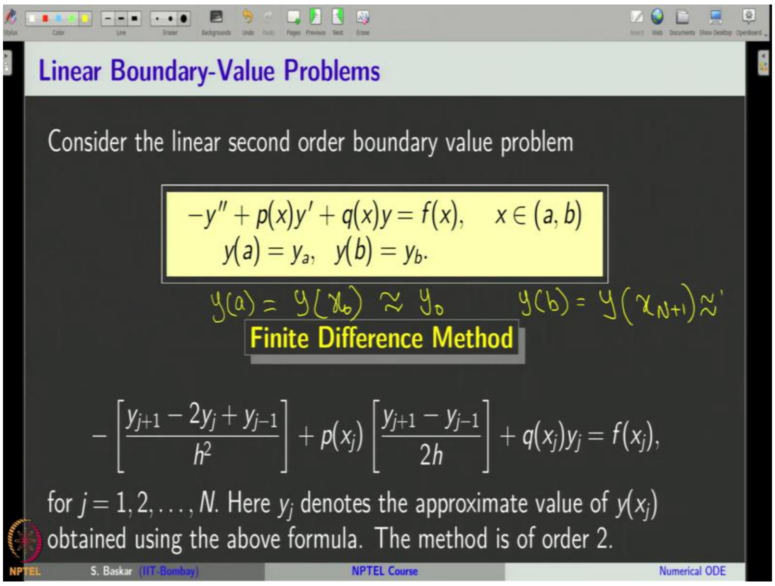
Numerical Analysis
Prof. S. Baskar
Department of Mathematics
Indian Institute of Technology - Bombay

Lecture - 61
Numerical ODEs Two-Point Boundary Value Problems

Hi, we are discussing numerical methods for ordinary differential equations. So, far we have discussed numerical methods for initial value problems. In this lecture we will discuss some numerical methods for 2-point boundary value problems. Let us start our discussion with linear problems.

(Refer Slide Time: 00:40)



Consider the linear second order boundary value problem given by $-y'' + p(x)y' + q(x)y = f(x)$, where p, q and f are given continuous functions and the problem is posed on an interval (a, b). We are also given 2 conditions now, one is at the point $x = a$, at this point the function y takes the value y_a and at the point $x = b$ the function y takes the value y_b.

And we want the solution y in such a way that it satisfies these 2 conditions at the boundaries of the interval, and at all the interior points it has to satisfy this second order linear ordinary differential equation. Since the conditions are specified at 2 different points, the problem is said to be a boundary value problem. As we did in the initial value problems, the first step is to get a discretization of the partition for the given interval $[a, b]$.

Let us partition the interval into sub intervals, with the end points as $x_0, x_1, \cdots, x_{N+1}$, where each x_j is given by $x_0 + jh$ where $h = \frac{b-a}{N+1}$. We will always assume that a unique solution for the given boundary value problem exists and it is sufficiently smooth. The first step for us to devise a numerical method for this problem is, to use the Taylor's theorem to approximate the derivative y'.

You can see that we are using the central difference formula to approximate y' therefore it is given by $\frac{y(x_{j+1})-y(x_{j-1})}{2h}$ and you have the remainder term, no\ow given at some point ξ_j where ξ_j lies between x_{j-1} to x_{j+1}. Similarly, you can get an approximation for y'' also. We will take again the central difference approximation for y''.

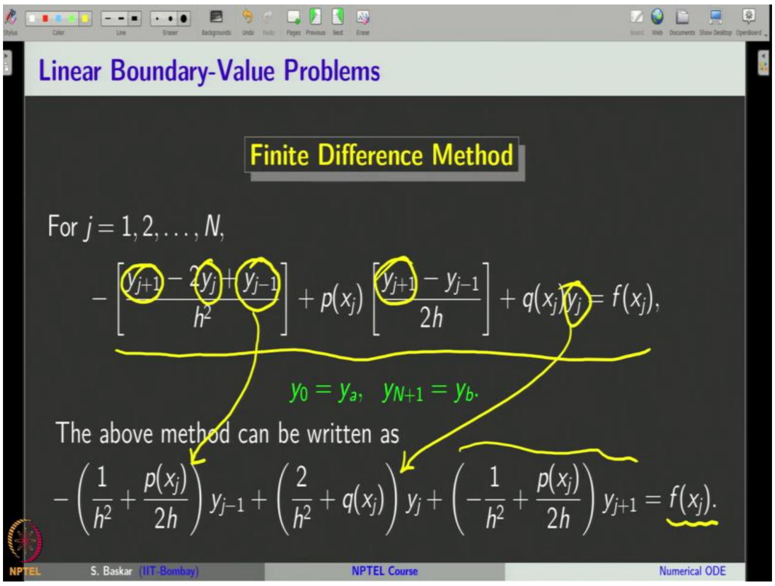
If you recall in one of our previous lectures on finite difference formulas, we have derived this formula using method of undetermined coefficients. We have also derived this truncation error there and therefore this formula is familiar to us. Now what we will do is, we will replace y'' by this central difference formula and y' by the central difference formula for y' and thereby the error that we are committing is of order 2.

Remember, we have already divided by h and h^2 for y' and y'' respectively. Therefore, when you replace the central difference formulas in the equation, you get a numerical method which is of order 2 now. So let us do that, when you replace y'' and y' by their corresponding central difference formulas, we get this equation. This is the central difference method for the second order ODE given here.

Where we have replaced the central difference formula for y'' in the first term and we have replaced the central difference formula for y' in the second term and this holds for all $j = 1, 2, \cdots, N$. Here, we have used the notation y_j to represent the approximate value of the solution at $y(x_j)$. As we told in the previous slide, the order of the method is 2. It is a second order method.

Remember, we also have the boundary conditions at $y(a)$. Remember $y(a)$ is nothing but $y(x_0)$, that we have indicated by y_0. Similarly, $y(b) = y(x_{N+1})$
and that we use the notation y_{N+1}.

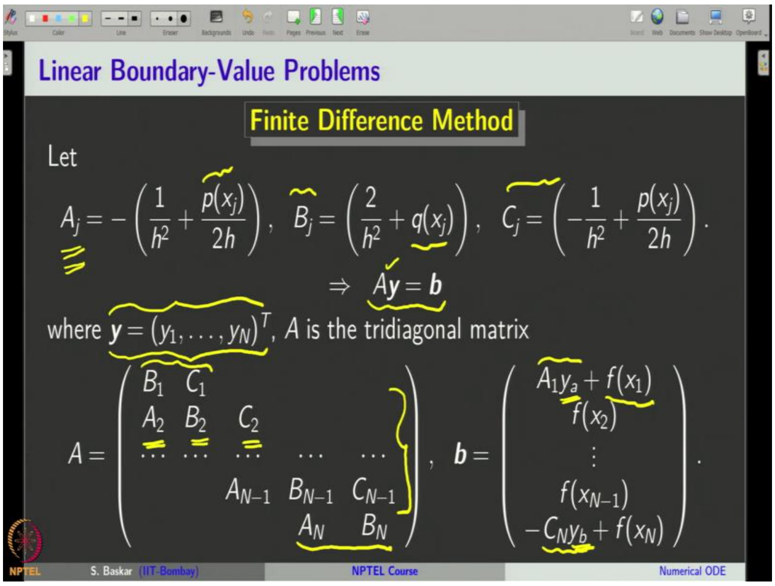
(Refer Slide Time: 05:57)



Therefore, we have this finite difference method together with the boundary conditions $y_0 = y_a$ and $y_{n+1} = y_b$. Remember, we are not making any approximation at $j = 0$ and $j = n + 1$. They are directly taken from the given boundary conditions. If there is no rounding error, these are exactly represented. Let us rewrite this expression in a different way. Let us collect all the terms of y_{j-1} and keep them together.

Similarly, all the terms of y_j are gathered and kept as the second term and finally all the terms of y_{j+1} are grouped and kept as the third term and of course we have the right hand side. So, we just rearranged the terms and we considered the finite difference method in this form.

(Refer Slide Time: 07:09)



Now let us see how this equation looks like, when you put $j = 1$. Remember, this is the finite difference method that we have after rearrangement and this holds for $j = 1,2,\cdots,N$. For $j = 1$, you can see that the first term is this, with $j = 1$ here. That is what we are writing here and then y_{j-1} becomes y_0. Similarly, the second term is the same here only thing is, now we have put $j = 1$ here and y_j is now becoming y_1.

And similarly, the last term is the coefficient with y_2. Remember p, q and f are given functions in our problem therefore all these terms are known to us and also $f(x_1)$ is known to us. Now look at the first term, the first term is appearing with y_0, which is also known to us from the boundary condition. Therefore, the first term is fully known to us. And so, you can simply push the first term to the right hand side and write the equation for $j = 1$ as this into y_1.

Remember y_1 is unknown here, this is known and y_2 is unknown, equal to this full term is known to us. thanks to the boundary condition because of that y_0 is also known to us. Similarly, let us take $j = N$ and then you write this equation. You can see that the first term is given by this expression into y_{N-1} which is unknown + this expression with $j = N$ into y_N. Again, this is unknown and the last term is this expression with $j = N$.

And now we have y_{N+1}. Again, you can observe that y_{N+1} is the right side boundary condition and therefore this is also known to us. So, you can just take it to the right hand side and write the equation for $j = N$ as the coefficient with y_{N-1} which is unknown and then the corresponding coefficient with y_N which is also unknown equal to the full known quantity where y_{N+1} is now known to us from the right side boundary condition.

Therefore, we have a set of equations like this where the first equation has 2 terms on the left hand side because we have pushed the first term to the right hand side because of the boundary condition. Similarly, the last equation has only 2 terms on the left hand side and the right hand side has 2 terms where this term coming from the boundary condition. All the interior points that is $j = 2,3,\cdots,N-1$, you have 3 terms for y_{j-1}, y_j and y_{j+1}.

So, for each j you have the diagonal term, the lower diagonal term and the upper diagonal term and all the coefficients are known to us. You can see that the unknowns are $y_1, y_2, \cdots, y_N$ and all these coefficients are known. You can observe that this forms a linear system which can be written as $A\boldsymbol{y} = \boldsymbol{b}$, where $\boldsymbol{y}$ is the approximate solution for our boundary value problem given by $y_1, y_2, \cdots, y_N$.

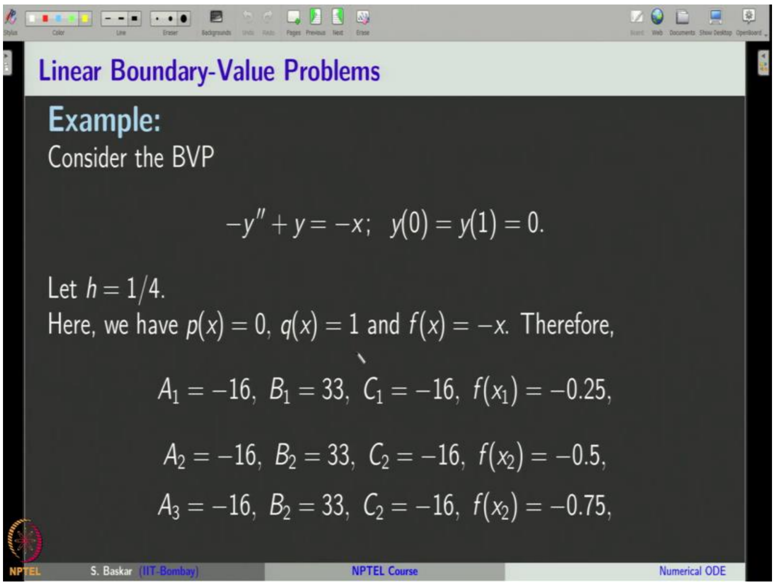
And what is this coefficient Matrix A? Well, the coefficient Matrix A is coming from these terms. Let us use a notation A_j to denote this term, that is the lower diagonal term. Let us use the notation B_j for the diagonal coefficient and C_j for the upper diagonal coefficient. In that way you can see that the first equation has only 2 terms because you have pushed the boundary term on the right hand side.

Similarly, the last equation has only 2 terms where again the boundary term from the right side boundary is pushed to the right hand side. All the interior equations have 3 terms, the lower diagonal term, diagonal term and the upper diagonal term. In that way, we got a tri diagonal system. If you recall, we have discussed an algorithm to solve a tri diagonal system, it is Thomas algorithm.

You can use Thomas algorithm to obtain the solution, which is the approximate solution of your given linear boundary value problem. You can immediately see that you can develop a python code to obtain the approximate solution of the given linear boundary value problem. Once you are given y_a, y_b, which are the boundary conditions, p, q and f which are the coefficient functions.

You can generate the elements of the Matrix A which are A_j, B_j and C_j. You can send this information into the subroutine for Thomas algorithm and you can get the unknown vector $\boldsymbol{y}$ which is the approximate solution of the linear boundary value problem. I hope you can code this method.

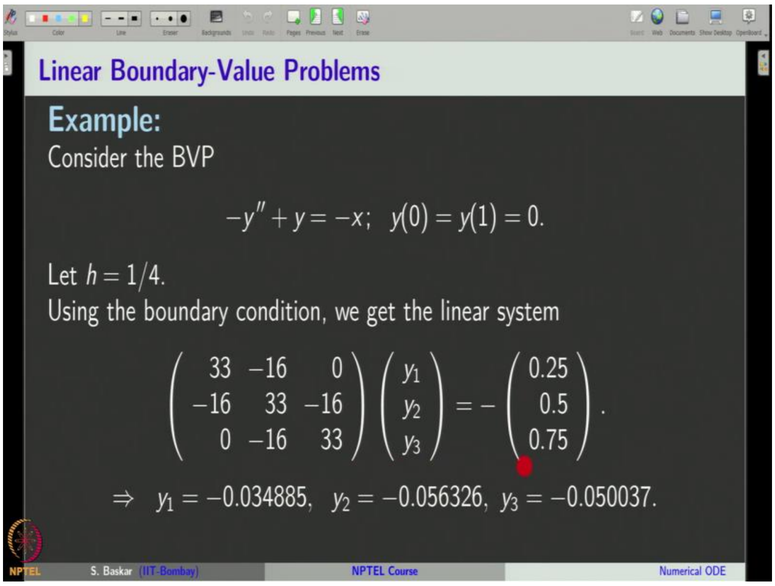
(Refer Slide Time: 13:56)



Just to have a clarity, let us take a very simple boundary value problem and compute the solution manually and see how it works. Let us take $-y' + y = -x$. This is our equation, we are given the homogeneous boundary condition $y(0) = y(1) = 0$. Let us take $h = 1/4$, you can see from the boundary condition that we are interested in solving this problem in the interval $[0,1]$ and we have taken the step size as 1/4. By this you can see that we will have a 3×3 tri-diagonal system. Here, $p(x) = 0, q(x) = 1$ and $f(x) = -x$.

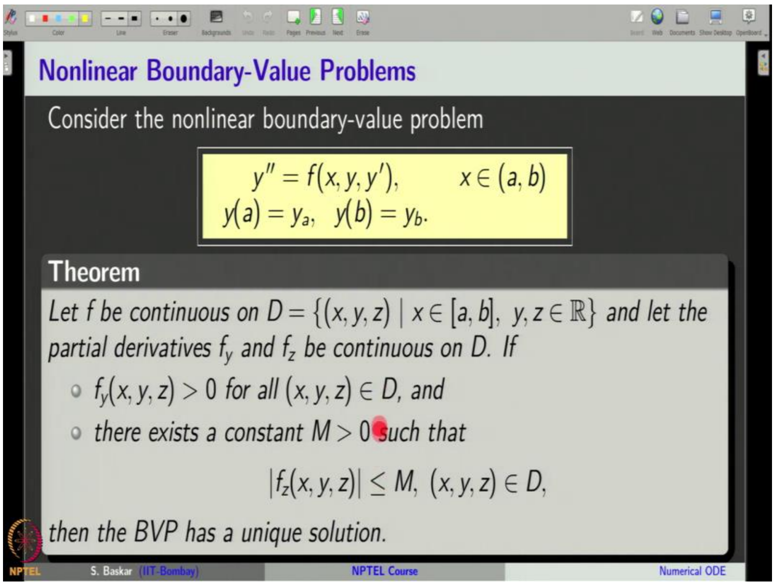
You can compare this equation with the given general equation of our problem. From there you can see this information. Once you have this information, you can go back to the definition of A_j, B_j and C_j and you can compute their values for each j. Since p and q are constants, you can see that A, B and C are constants, that is they do not change for different j's, only f will change and they are given by these values.

(Refer Slide Time: 15:32)



Once you have all this information, you can immediately write the tri diagonal system $A\boldsymbol{y} = \boldsymbol{b}$ and it is given by this. In fact, you can use the Gaussian elimination method also to solve this system. You can see that the solution is given like this.

(Refer Slide Time: 15:51)



So, with this we have completed the finite difference method for linear boundary value problem. Let us move on to non-linear boundary value problems. Consider the non-linear boundary value problem $y'' = f(x, y, y')$. In general, f can be a non-linear function of y and y'. In which case, the boundary value problem will be a non-linear boundary value problem.

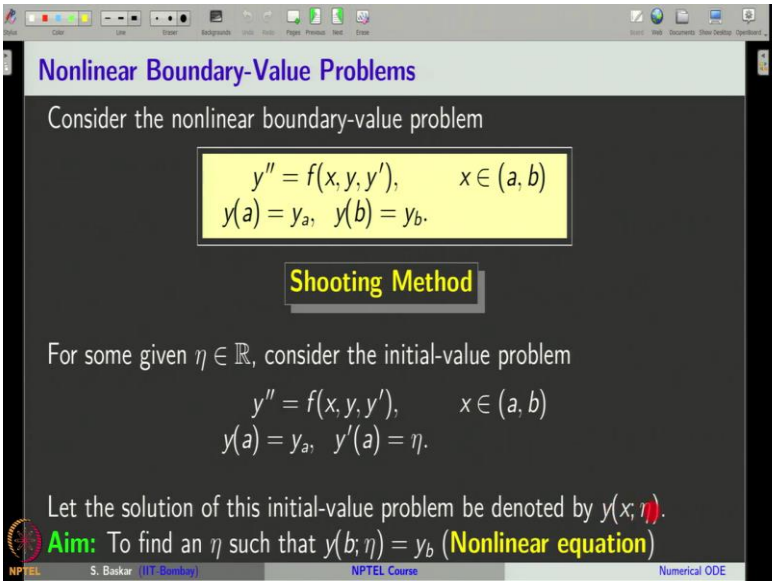
And it is posed on the interval (a, b), with the conditions that $y(a) = y_a$ and $y(b) = y_b$. y_a and y_b are given to us. y_a and y_b are some real numbers. Therefore, we have a boundary value problem again. Now it is a non-linear boundary value problem. You can also apply the finite difference method, that we have discussed in the previous section. But in that case, you may land up with a non-linear system of equations, for which you may have to use the Newton's method, that we discussed in the non-linear chapter.

But here we will use another interesting method, called Shooting method. Before getting into the method, let us quickly see the existence and uniqueness theorem of this non-linear boundary value problem. We will only state the theorem but the proof of this theorem is not the subject of numerical analysis. Let *f* be a continuous function defined on the domain $D = \{(x, y, z) | \; x \in [a, b] \,, y, z, \in \mathbb{R}\}$.

We assume that the partial derivative of *f* with respect to *y* and *z* be continuous on the domain *D*. And further, if $f_y(x, y, z)$ is positive for all $(x, y, z) \in D$ and $f_z(x, y, z)$ is bounded in the domain *D* then a unique solution for the given boundary value problem exists. Therefore, when we devise the numerical method, we have to make sure that all these conditions are satisfied by the right hand side function that we are considering.

Otherwise, what happens is you may be devising a numerical method for which there may not be any solution. In order to avoid such a situation, it is more safe for us to work one with those functions *f* that satisfies all this hypothesis. Anyway, this is just a mathematical remark.

(Refer Slide Time: 18:52)



Let us go to the method that we are interested in, it is called the Shooting method. Main idea of the Shooting method is, to first choose some η, you may choose it arbitrarily and consider the initial value problem $y'' = f(x, y, y')$ posed on the interval (a, b). Remember that, this is the same as the equation that you have in your original boundary value problem and now you have initial conditions.

One initial condition is the same, as it is given in your original problem, that is $y(a) = y_a$. Therefore, you are having the left boundary condition from your boundary value problem fixed as the condition at $x = a$ in your initial value problem also. Now you have one more condition because you have second order equation. Therefore, you need 2 conditions in order to have a unique solution.

For that reason, you need one more condition in our original problem. We have specified that additional condition at the point b, therefore it became a boundary value problem. Now we are fixing that additional condition at the point $x = a$ itself. But now, we are specifying the condition at y' and we take $y'(a) = \eta$. Remember η is something that we arbitrarily choose and then fix it here.

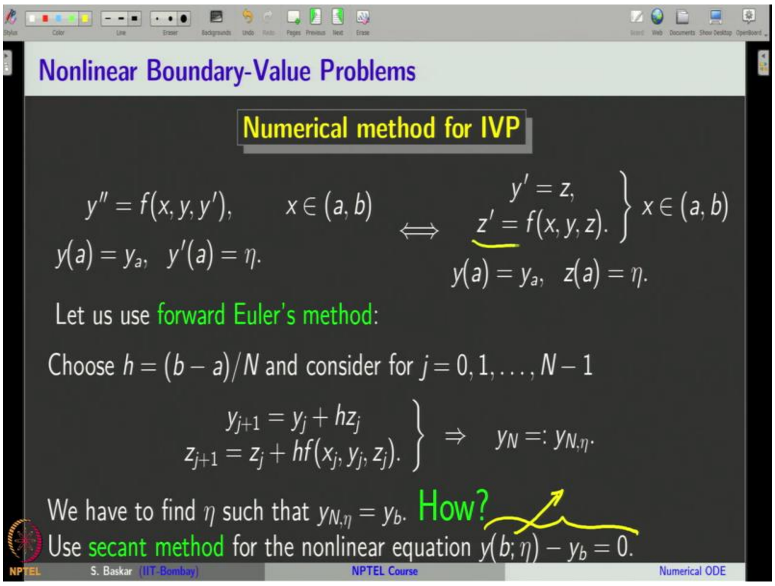
Now if you can solve this initial value problem and get the solution, then we will denote that solution as $y(x; \eta)$, where η is the parameter and *x* is the independent variable in your problem. An interesting observation here is, to see that if you chose η in such a way that $y(b; \eta) = y_b$ then that *y* will also be the solution of your boundary value problem. Why? Because your *y* already satisfies your equation and also it satisfies the left side condition $y(a) = y_a$.

Now if you have chosen your η such that $y(b; \eta) = y_b$, then you are completely done with your boundary value problem solution also. But we really do not know what is that η which gives you the solution *y* such that $y(b; \eta) = y_b$. We really do not know this, therefore in general, this will define a non-linear equation whose solution is precisely the η for which you have $y(b; \eta) = y_b$.

So, the idea is to choose an η, solve this initial value problem, get the solution *y* and then plug in that *y* into this equation and solve this non-linear problem to get the η. Remember this is the equation with variable as η therefore its root will precisely be the value of η at which $y(b; \eta) = y_b$. So, how are we going to achieve this? Well, you can use any non-linear iterative method that we have introduced in one of our previous chapters to solve this non-linear equation.

In our case, we will use the Secant method but before showing you how to set up the Secant method, let us first worry about how to solve this initial value problem and get this solution. Because unless you get the solution, you cannot go to set up the Secant method to solve this non-linear problem. Therefore, let us first go to solve this initial value problem.

(Refer Slide Time: 23:25)



Well, you can solve this initial value problem in 2 ways. One is, you can replace y'' by the central difference formula and replace y' by again its corresponding central difference formula but that may give you a system of non-linear equations. Therefore, another nice way to get the solution of the second order initial value problem is, to convert the second order equation to a system of first order equation.

How can you do that? If you would have done a basic course in ODE, you would have learned that any higher order equation can be converted to a system of first order equations. Here you have second order equation therefore, when you convert it to a first order equation, you will get 2 first order equations. The idea is very simple even if you have not done a course on ODE. It is not very difficult for you to understand this.

What you do is, first you define a function z which is equal to y'. Once you have this then what is z'? z' is nothing but y'' and y'' is from your original equation is given by $f(x, y, y')$. That is what we are writing here, z', which is actually equal to y'', $= f(x, y, z)$, instead of y' we will put this z here. In that way, you have cleverly eliminated y'' and got this system of 2 equations, that involves only the first order derivative of the unknown variable.

Now instead of posing the second order equation on the interval $[a, b]$, now we will pose this system of first order equations on the interval $[a, b]$ with the same initial condition, $y(a) = y_a$ and instead of $y'(a)$, now we will put $z(a)$ because that is the notation we are using here. Therefore, it is $z(a) = \eta$. Now you can see that you have 2 first order equations. You can use any numerical method that we have developed for first order initial value problem.

Now you see, we have a system of first order equation with initial condition. Recall, we have developed many methods to approximate solution of a first order initial value problem. You can use any of those methods, like forward Euler method or Runge-Kutta method or any other multi-step implicit or explicit methods to approximate solution of this system of first order equations. Only thing is, you have to apply that method twice.

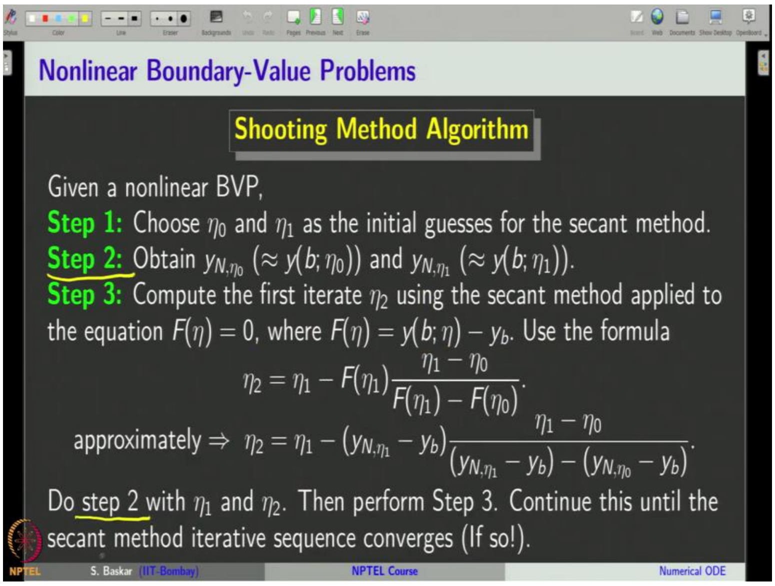
One for the first equation and another for the second equation. To illustrate that, we will consider this simplest possible method that is the forward Euler's method and see how to implement the forward Euler method for this system of first order equations. There is nothing difficult here, you just have to apply the forward Euler method individually to both this equation. Choose $h = \frac{b-a}{N}$ and consider for $j = 0,1,2,\cdots, N-1$.

The Euler forward method for the first equation, which is given by $y_{j+1} = y_j + hz_j$ and similarly you have the forward Euler method for the second equation given by $z_{j+1} = z_j + hf(x_j, y_j, z)$. Now you will see, starting from $j = 0$, you can go up to $j = N-1$. At $j = N-1$, you have y_N, that is precisely the approximation for $y(b'\eta)$ for a given η.

So, that is approximately equal to y_N that you obtained using the Euler method. So, to obtain that you also need to find z_{j+1} for every j because it is a coupled system. So, once you have y_N, let us denote it by $y_{N,\eta}$ because we are choosing an η and then setting up this Euler method and computing y_N. Therefore, your y_N will surely depend on the choice of η that you have taken at the beginning of this problem. Therefore, we will use the notation $y_{N,\eta}$.

Now once for given η, you know how to get $y_{N,\eta}$. Now, we want that η for which $y_{N,\eta} = y_b$. So that is what ultimately we want to have, but unfortunately we do not know that value of η. Now how to get that? We will try to capture that η through an iteration. We will use secant method to solve this non-linear equation by replacing $y(b;\eta)$, which is the exact solution of this system, now by the approximate solution $y_{N,\eta}$.

(Refer Slide Time: 29:32)



Now we have gathered all the information to set up the secant method. Let us now write the algorithm in a systematic way and call it as the Shooting method. Remember we are given a non-linear boundary value problem. What is that problem? That problem is here, once you are given the non-linear boundary value problem, you first have to set up an initial value problem, for that you will choose an η.

Now remember, we want to use secant method. In secant method, you need 2 initial guesses. Therefore, we will choose η_0 and η_1, two initial guesses for this secant method. For η_0, we will first solve the initial value problem, that is this initial value problem with $\eta = \eta_0$. How we are doing it? We are first converting it to a system of first order ODE and then using the forward Euler method to get y_{n,η_0}.

Similarly, you give $\eta = \eta_1$. Correspondingly, you set up the initial value problem for the first order system. Use forward Euler method to get y_{N,η_1}. So, therefore once you choose η_0 and η_1, you can get y_{N,η_0} and y_{N,η_1}. These are the approximations of the solution of your initial value problem with η_0 and η_1 as parameters. Once you have these 2 values, you are now ready to set up the secant method.

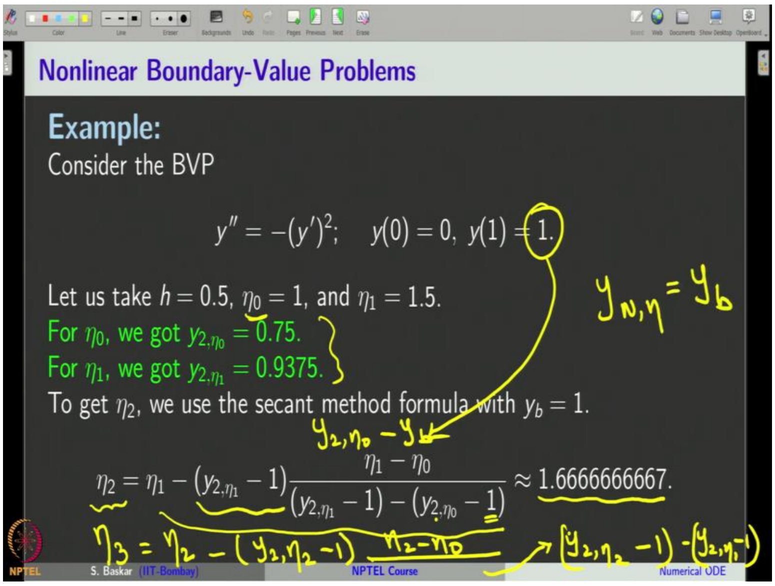
Remember, you have to apply the secant method for the non-linear equation $y(b;\eta) - y_b = 0$. Only thing is, instead of exact solution coming from your initial value problem, now you will plug in these approximate solutions. Thereby, you are supposed to set up this as the iterative formula coming from the secant method applied to this equation.

You should go back to our non-linear equations chapter. Recall the formula for secant method for a given non-linear equation and come back and see this is the formula. Here, $f(\eta)$ is given by this. Now what you will do is, instead of using the exact value, now you will use the approximate value that is $y_{N,\eta}$, computed using the Euler method. Similarly, you can also apply Runge-Kutta method or any other method, trapezoidal method or any predictor corrector method.

Anything you can use to approximate these solutions here. To have simplicity, we have used forward Euler method and similarly you also have y_{N,η_0} and you can plug in these values into the secant method formula and get η_2. Now once you have η_2, again you set up the initial value problem, that is you go back to the step 2, you set up the initial value problem here.

Now with η_2 and again you do this process get y_{N,η_2}. Once you have y_{N,η_2}, you can use these 2 values to get η_3 from the secant method and like that you can keep on iterating ηs. And if this sequence converges, you will eventually get that η for which $y(b;\eta) = y_b$. So, that is the idea of the Shooting method.

(Refer Slide Time: 33:59)



Let us illustrate the Shooting method with this simple non-linear ODE. We have $y'' = -(y')^2$. It is a non-linear ODE and we are given the boundary conditions as $y(0) = 0$ and $y(1) = 1$. Let us take $h = 0.5$ because we are going to do the computation manually. Therefore, it is better to take some big head h. So, that there is no much computation involved. To start with, we have to choose 2 initial guesses η_0 and η_1, you can choose them arbitrarily.

Here I have chosen them as $\eta_0 = 1$ and $\eta_1 = 1.5$. The first step is to take η_0 and use the numerical method to solve the initial value problem. We decided to take Euler method, that is forward Euler method, therefore we will apply the forward Euler method to the initial value problem. With initial condition as $y_0 = 0$, that is given here, and $y'(0)$ that is now z_0.

Because in our initial value problem, we have taken $z = y'$, that is why $z_0 = y'(0)$ and that we will take as η_0, the parameter that we have chosen. So, once you have this initial value problem, you will go to apply the Euler method component wise, that is for the equation $y' = z$. That gives you y_1 and then you apply the Euler method to the second component that is $z' = f(x, y, z)$.

And that gives you z_1 for the first step. Now once you have the first step you have to go for the second step that is $j = 0$ gives you this. Remember y_1 is the approximate solution for $y(x_0 + h)$, that is y of, x_0 is zero, therefore 0 + h is 0.5. Therefore, it is 0.5. This y_1 is the approximation of $y(0.05)$. Similarly, you need to have y_2 which is an approximation for $y(1)$ and that is the right hand side boundary.

Therefore with this h, you just have to go 2 steps in the Euler method. y_2 is given by 0.75, you can check that and that is precisely what you want to have as the value for the right side boundary with η_0 as the parameter right. Well, you may have to also find z_2 but that is not required because for our Shooting method we only want this value. Once you have this value, you do not need to find z_2 but you need to find z_1 because that is plugged in the expression for y_2.

Therefore, you need to find z_1. Once you have z_1, you can compute y_2. Once you have y_2, your purpose is achieved for this particular η_0. Therefore, you need not compute z_2. Now let us take η_1. Remember we have already computed y_{2,η_0} corresponding to η_0. Now we have to find y_{2,η_1}, corresponding to the parameter η_1. For that, again we will apply the Euler method with the initial condition as $y_0 = 0$ and $z_0 = \eta_1$.

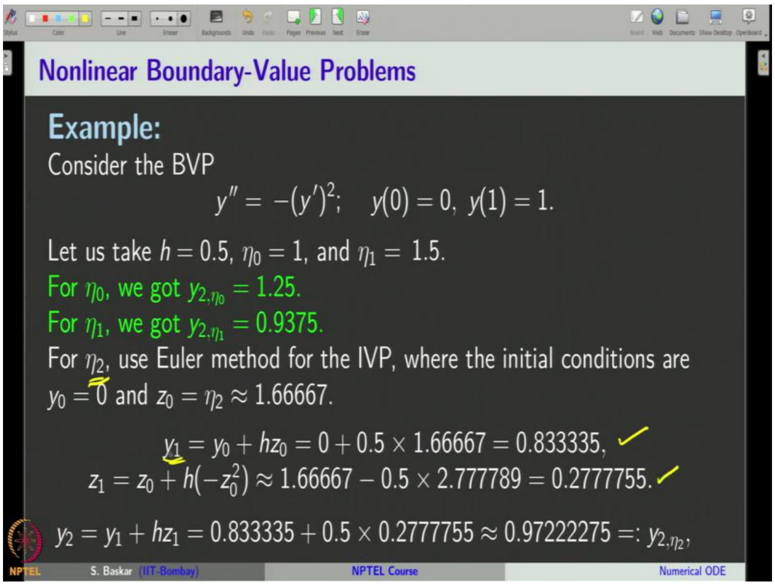
You can again find y_1, z_1, plug in y_1 and z_1 into y_2 and get y_2 as this and denote it by y_{2,η_1}. Remember, you already got y_{2,η_0}. Now you got y_{2,η_1}, therefore you are now ready to apply the secant method to get η_2. And that is given by this formula and when you plug in all these values into η_2, remember, we have $y_{2,\eta_0} - \mathrm{y_b}$. What is y_b? y_b is 1, that is why we have here $y_{2,\eta_1} - 1$ and similarly $y_{2,\eta_0} - 1$ here and that reduces to this value and that is your η_2.

Once you have η_2, again you go back to the Euler method with η_2 and compute y_1, z_1 and then y_2 corresponding to η_2. Once you have y_{2,η_2}, you can come to compute η_3 which is equal to $\eta_2 - \left(\mathrm{y}_{2,\eta_2} - 1\right)\frac{\eta_2-\eta_1}{(\mathrm{y}_{2,\eta_2}-1)-(\mathrm{y}_{2,\eta_1}-1)}$. So, that will give you η_3. Again, once you have η_3, you will then go to the Euler method to compute y_{2,η_3}. Like that the iteration will keep on going.

It may be little confusing but you have to carefully understand it. Once you understand, it is very clear how the iteration goes. First set up the initial value problem and then solve that initial value problem using Euler method or any other method. For setting up the initial value problem, you need to choose the η first time, η_0 and η_1 and once you have the corresponding *y* values on the boundary then you will come back to the secant method and get the next iteration for η.

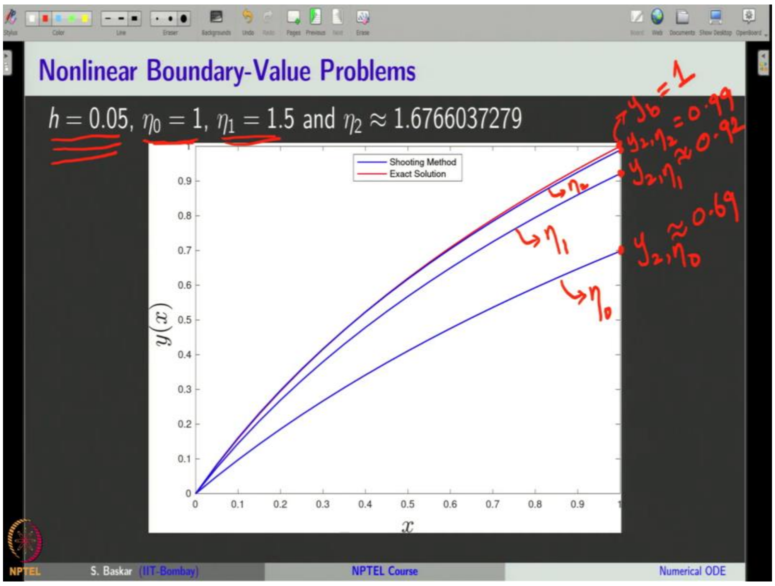
Like that the iterative sequence will go and if the secant method iteration converges then this η which comes as the limit of the sequence will be the η for which you have $y_{N,\eta} = \mathrm{y_b}$.

(Refer Slide Time: 41:31)



So, let us see how this solution looks like. For η_2, you have y_1 is equal to this value, z_1 is equal to this value and once you have y_1 and z_1 you will plug in to y_2 to get y_{2,η_2}. Once you have y_{2,η_2}, you can go to compute η_3 and the iteration goes on like this.

(Refer Slide Time: 42:09)



Let us see how the graph of the solution looks like. Here the red line represents the exact solution of the given boundary value problem and the blue line gives the approximate solution for each η computed using the Euler method. The first line corresponds to η_0 and for that y_{2,η_0} is roughly 0.75 and this is the solution computed for η_1. Remember, you have y_0, y_1 and y_2, which we denote by y_{2,η_1}.

Similarly for the η_2, you have this graph and this is the solution obtained using the Shooting method. You can see that as you go on with η in the secant method, your $y_{2,\eta}$ is going more and more closer to the exact boundary condition, this is y_b for you. Remember, you have taken y_b as 1 right. So, it is trying to approach this condition, of course the solution otherwise is not. So, good because we have taken $h = 0.5$. If you take h to be something more smaller say 0.05 and all the approximation will be better.

But we cannot do it manually. You can develop a python code and compute the solution for smaller values of h. Here, I will show you the solution computed using the Shooting method with $h = 0.05$. You can see that η_0 we have taken as one as usual and η_1 is taken as 1.5. For η_0, the solution is this and you have y_{2,η_0} is something approximately 0.69.

And this is the graph corresponding to η_1 and that is giving us the boundary condition as y_{2,η_1}, which is approximately 0.92 and this blue line corresponds to η_2 and that gives us the boundary condition as y_{2,η_2} is almost 0.99. You can see if you take η_3, η_4 and all, it will try to approach the exact boundary point which is $y_b = 1$. So, this is how the Shooting method works. It may be little confusing but if you carefully understand, it is not very difficult for you to code it.

Given that we have already learned the coding for Euler method in the last class, you can now combine the Euler method and the secant method and I hope you can develop a python code for the Shooting method also. With this note let us end this lecture. Thank you for your attention.