**Numerical Methods for Science and Engineering**

**Ordinary Differential Equations**

**1. Introduction**

We shall consider the solution of ordinary differential equations satisfying certain conditions. Problems in which all the initial conditions are specified at one point only are called *initial value problem* (I.V.P.). On the other hand, problems involving second and higher order differential equations, we may prescribe the conditions at two or more points. Such problems are called *boundary value problems* (B.V.P.).

Any higher order differential equation can be expressed as a system of first order equations and the method of solution for first order equation can extended for the system.

To describe various numerical methods for the solution of ordinary differential equations, we consider the general first order differential equation:

 (1)

with the initial condition



**2. Taylor Series Solution**

Consider the Taylor series solution of the of the initial value problem:

with

The solution of the equation is a function of *x*. The Taylor series expansion of  about  is,



where is the value of  at .

The values of the derivatives can be found by differentiating repeatedly and substituting and .

The method can also be used for higher order differential equations.

**3. Euler Method**

Consider the solution of the initial value problem,

with (1)

From Taylor series expansion of and neglecting terms containing  and higher powers of *h*, we have



or equivalently



This is Euler’s formula, and the order of the error is .

Alternatively, integrating (1) from *x*0  to*x*1 with respect to *x*, we have



or 

Assuming that in , we have

where

The process can be continued for the next interval by taking  and  as the starting values. In general , we have the iteration formula

 

**4. The Modified Euler’s Formula and Predictor-Corrector Method (Omitted)**

Integrating the DE (1) from *x*0  to*x*1, we have



Using trapezoidal rule to the integration we obtain the modified Euler’s formula



An iterative formula to the above equation can be taken as



The iteration can be started by choosing from the Euler’s formula



This procedure is known as one-step ***predictor-corrector*** method.

**5 The Runge-Kutta Method**

**5.1 Second Order Runge-Kutta Method (RK-2 Method)**

If we use  on the right side of the modified Euler’s formula, we have



Setting





we have 

This is known as the second order Runge-Kutta formula.

It can be shown that RK-2 method is equivalent to Taylor series of order two and the order of the error is 

**5.2 Fourth Order Runge-Kutta Method (RK-4 Method)**

The fourth order Runge-Kutta formula, the most commonly used one in practice, is stated without proof.



where









Note that the process is not unique, and many other variations are possible. In fact the fourth order process is very accurate and most frequently used. This formula is equivalent to Taylor series of order four and the order of the error is .

**6 System of Equations**

So far we have considered the method for a single differential equation but those methods can be extended for system of first order equation.

Consider a system of pair of equations





subject to initial conditions  and .

**(a) Euler’s Method for a System**

Euler’s method to the above system can be written as





**(b)Rk-2 Method for a System**

The second order Runge-Kutta (RK-2) method to the above system can be written in the form

 

 

and 



In a similar way RK-4 method can be extended for system of equations.

An ordinary differential equation of order higher than 1 can be solved numerically by changing it into a system of first order equations.

**7 Higher Order Differential Equations**

The Taylor series solution for higher order differential equations is straightforward and similar to first order equation.

An ordinary differential equation of order higher than 1 can be solved numerically by changing it into a system of first order equations. Consider, for example, a second order initial value problem

 with and 

Defining the new variable



the above initial value problem can be written as





with and .

**Example 1 :**Given that , where .

(a) Estimate the values of using Euler’s method with step size .

(b) Estimate the values of using the Runge-Kutta method of order two.

(c) Estimate a value of using theRunge-Kutta method of order four.

(d) (i) Use three point central difference formula for derivative to derive a recurrence relation for the above IVP.

(ii) Estimate the values of *y* at using .this recurrence relation.

(e) Use MATLAB function “**[x1, y1]=ode23(f, [x0, xn], y0)**”

(i) to estimate the values of *y* in using .

(ii) to plot the solution curve in .

**Solution (a)** Here and .

Taking , we have

(b) Taking and , , , we have







(c) Taking and , , , we have











(d) (i) Using three point central difference formula for derivatives with the differential equation at we have

(ii) For with the starting values , we get

Taking ,

Taking ,

(e) (i) >> clear all

>> f=@(x,y) 2\*x\*y^2-y;

>> [x1, y1]= ode23(f, [0:0.2:1.4], 1);

>> Solution = [x1,y1]

Solution =

0 1.0000

0.2000 0.8484

0.4000 0.7644

0.6000 0.7257

0.8000 0.7275

 1.0000 0.7799

1.2000 0.9253

1.4000 1.3401

(ii)

>> xin=0:0.02:1.4; % generate points with sm

>> [x2, y2]=ode23(f, xin,1);

>> plot(x2,y2)

**Exercise 2**

Given the initiaal value problem

,  with  and .

1. Estimate  and  using the RK-2 method with step size.
2. Use MATLAB function “**[x1, y1]=ode45(F, [x0, xn], y0)**”

(i) to estimate the values of *y* in using .

(ii) to plot the solution curve in .

**Solution**

1. Taking ,  and , ,  with , we have

|  |  |
| --- | --- |
|  |  |
|  |  |

Now starting with , ,  and , we have

|  |  |
| --- | --- |
|  |  |
| =2.7716 | = 2.7402 |

Thus

and .

(b) Using , we can write the system as follows:

Entry in command window;

(i) >> clear

>> F=@(x,y) [x+y(1)^2-y(2); x^2-3\*y(1)+y(2)^2];

>> [x1,y1]=ode45(F, [1:0.1:1.4], [2, 2.5]);

>> Sxyz = [x1,y1]

Sxyz =

1.0000 2.0000 2.5000

1.1000 2.3091 2.6241

 1.2000 2.8009 2.7112

1.3000 3.6825 2.6368

1.4000 5.6417 2.0508

(ii)

>> [x2,y2]=ode45(F, [1:0.01:1.4], [2, 2.5]);

>> plot(x2,y2)

**Example 3**

An LCR circuit can be described by the system of differential equations

where *I* is the current through the inductance and *V* is the voltage drop across the capacitor.

Suppose that *R*= 1 ohm, y.

Use MATLAB command “**[x1, y1]=ode45(F, [x0, xn], y0)**” to find the numerical solution for if and *V*

Plot your results and separately for distributions.

**Solution:**

Using , we can write the system as follows:



Entry in command window;

>> clear

>> F=@(t,y) [y(2); -y(1)-y(2)]

F = @(t,y)[y(2);-y(1)-y(2)]

>> [t,y]=ode45(F,[0,5],[2,3]);

>> plot(t,y(:,1))

>> xlabel('t'); ylabel('I(t)');

>> title('Distribution of I(t)');

>> plot(t,y(:,2))

>> xlabel('t'); ylabel('V(t)');

>> title('Distribution of V(t)');

**Example 4**: Consider the boundary value problem

1. Derive a recurrence relation using three points central difference approximations with
2. Using the above finite difference formula solve the above BVP.
3. Express the above initial value problem as a system of first order equations.
4. Use MATLAB function “**sol=bvp4c(odefun, bcfun, solinit)**” to the BVP.

(i) Estimate the values of *y* at . 1/3, 2/3, 0.4 and 0.9.

(ii) Plot the solution curve.*y*.

**Solution**

1. Using central difference formulas, we have

Taking , we get

On simplification,

1. With , the nodal points are .

For ,

For ,

Solving the equations (1) and (2), we have

1. Using , we can write the system as follows:

(d)

>> clear

>> F=@(x,y) [y(2); 1+y(2)]; % dy/dx = F(x, y)

>> bc=@(ya,yb) [ya(1)-1; yb(1)-2\*(exp(1)-1)]; % format boundary values

>> yinit=@(x) [1; 4]; % initial guess value supplied

>> solinit=bvpinit(linspace(0,1,3), [1,2]); % generates starting values

>> sol=bvp4c(F, bc, solinit); % solution

(i) % Extracting values from solution “sol”

>> xint=[0, 1/3, 2/3, 1];

>> sxint=deval(sol,xint)

Val\_xy =



0 1.0000

0.3333 1.4579

0.6667 2.2283

1.0000 3.4366

(ii) % Plotting of .

>> xval=linspace(0,1);

>> yval=deval(sol,xval);

>> plot(xval, yval(1,:))

**Exercise**

1. Given the initial value problem with .

(a) Estimate the values of using Euler’s method with step size .

(b) Estimate the values of using the Runge-Kutta method of order two.

(c) Estimate a value of using theRunge-Kutta method of order four.

(d) (i) Use three point central difference formula for derivative to derive a recurrence relation for the above IVP.

(ii) Estimate the values of *y* at using .this recurrence relation.

(e) Use MATLAB function “**[x1, y1]=ode23(f, [x0, xn], y0)**”

(i) to estimate the values of *y* in 1 using .

(ii) to plot the solution curve in .

2. Given the initial value problem

(a) Estimate the values of using the Runge-Kutta method of order two.

(b) Estimate a value of using theRunge-Kutta method of order four.

(c) (i) Use three point central difference formula for derivative to derive a recurrence relation for the above IVP.

(ii) Estimate the values of *y* at using .this recurrence relation.

(d) Use MATLAB function “**[x1, y1]=ode45(f, [x0, xn], y0)**”

(i) to estimate the values of *y* in using .

(ii) to plot the solution curve in .

3. Given the system

 and 

with the initial conditions and ..

1. Estimate  and  using Runge-Kutta method of order two.
2. Use central difference formula to derive a recuurence formula and estimate the values of .
3. Use MATLAB function “**[x1, y1]=ode45(F, [x0, xn], y0)**”

(i) to estimate the values of *y* in using .

(ii) to plot the solution curve in .

4. The equation for a circuit with applied voltage is

where and given that and at.

(a) Use three point central difference formula for derivatives to derive a recurrence relation for the above IVP.

1. Use (a) and (b) to estimate the values of *y* at t.
2. Express the above initial value problem as a system of first order differential equations.
3. Estimate a value of  using the Runge-Kutta mehod of order two.
4. Use MATLAB function “**[x1, y1]=ode45(F, [x0, xn], y0)**”

(i) to estimate the values of *y* in using .

(ii) to plot the solution curve in .

5. Consider the boundary value problem

1. Derive a recurrence relation for the above differential equation using three point central difference formula for derivative with .
2. Using the above finite difference formula solve the above boundary value problem.
3. Express the above initial value problem as a system of first order differential equations.
4. Use MATLAB function “**sol = bvp4c(odefun, bcfun,solinit)**” to solve the BVP.

(i) Estimate the values of .

(ii) Plot the solution curve ’

6. Consider the following boundary value problems

(i) .

(ii) .

(iii) .

Find the solution of the problem using followind steps.

1. Derive a recurrence relation for the above differential equation using three point central difference formula for derivative with .
2. Using the above finite difference formula solve the above boundary value problem.
3. Express the above initial value problem as a system of first order differential equations.
4. Use MATLAB function “**sol = bvp4c(odefun, bcfun, solinit)**” to solve the BVP.

(i) Estimate the values of .

(ii) Plot the solution curve ’

7. Solve the following boundary value problems using the finite difference method and central difference approximations.

(a) .

(b) .