Numerical Methods
For Solving
Higher-Order Boundary-Value Problems

By:

Basma Othman Al-Azzah

Supervised by:

Dr. Samir Matar

An-Najah National University

Faculty of Graduate Studies

Numerical Methods For Solving Higher-Order Boundary-Value Problems

By:

Basma Othman Al-Azzah

Supervised by:

Dr. Samir Matar

Submitted in Partial Fulfillment of the Requirements for the Degree of Master in Computational Mathematics, Faculty of Graduate Studies, at An-Najah University, Nablus, Palestine.

Numerical Methods
For Solving
Higher-Order Boundary-Value Problems

By:
Basma Othman Al-Azzah

This Thesis was defended successfully on 6 / 9 / 2009 and approved by

Committee Members	**Signature**
1. **Dr. Samir Matar (Supervisor)**	…………………
2. **Dr. Mohammed Najib Ass'ad (Internal Examiner)**	…………………
3. **Dr. Sae'd Mallak (External Examiner)**	…………………

Dedication

To My Inspirer, My Mother, My Family and To My Dad Soul for Their Support, encouragement and Invocations.

Acknowledgments

First, my greatest thanks for my God for helping me finish this work as good as I hope.

Then my all thanks and wishes for

My supervisor Dr. Samir Matar who directed me to finish this research successfully.

Dr. Mohammad N.Ass'ad who supported and helped me to be one of researcher students and granted me his trust.

Finally, my all thanks and wishes for my mother especially, my little teacher -my sister- who taught me the basic rules of programming, my family, and to my friends for all kinds of support and concern.

Contents:

Abstract

The Boundary Value-Problems (BVPs) either the linear or nonlinear problems have many life and scientific applications.

Many studies concerned with solving second-order boundary-value problems using several numerical methods, and few studies concerned with especial cases of higher order boundary-value problems using several numerical methods to solve them. But However, in our thesis we concerned with the finite-difference methods for solving general high-order linear boundary-value problems (from order three up to order seven), modifying, and developing some finite-difference methods for solving especial eighth-order nonlinear boundary value problems to enable them solve any even-order problem beyond it.

The main steps in this thesis depended on:

- Using special finite-difference approximations for derivatives and formatting a formula that can be deal with endpoints that exceed the usual finite-difference formula for derivatives
- Constructing linear system and solved it using the LU-decomposition method to decrease the computational processes.
- Using The Richardson's Extrapolation method to get more accurate results.

The numerical results for methods that appear in this thesis are good, but the errors in the methods increase when the order of the boundary-value problems becomes higher i.e. the fifth-order problem needs all derivatives

$y', y'', \dots, y^{(iv)}$, so by using approximations many times the errors will increase. Also the method accuracy depends on the boundary conditions values, that method has large error when α_0 , β_0 are not given nor one of them, as well as the changes on the order of boundary conditions values. Besides that, this method requires long computing time and hard work while using very large equations that increase while the problem order increases.

CHAPTER ONE

Preview

In this chapter we will discuss some important definitions that are necessary in our research, and very important three new algorithms used to find the finite- difference approximation for any derivative with associated local truncation error.

The main idea for two of these algorithms depends on the undetermined coefficients method that is using a set of points or derivatives.

The first algorithm in this chapter is used to derive the finite-difference approximation for any derivative, and the second is used to derive a finite-difference approximation for any derivative using a set of points and a set of known derivatives (The boundary conditions for problem).

The third algorithm is used to find local truncation error for finite-difference approximation for any derivative that is derived using algorithm one and two and determined its order.

The importance of these algorithms came from our need to use them many times with many permutations until getting suitable formula and convergence without hard hand work.

1.1 Definitions:

In this section we will demonstrate some important definitions for terms that we will use in our thesis.

Ordinary Differential Equation (ODE):

It is a relation that contains functions of only one independent variable, and one or more of its derivatives with respect to that variable. See [4]

The Order of Differential Equation:

It is the order of the highest –order derivative involved in the equation.

Example: $\frac{dy}{dx}+\frac{d^3y}{dx^3}=e^{-x}$ is a Third-Order ordinary differential equation. See [8]

The Degree of the Differential Equation:

It is the highest power to which the highest-order derivative is raised.

Example:

$(\frac{\mathrm{dy}}{\mathrm{dx}})^5+(\frac{d^3y}{dx^3})^2-xy=e^{-x}$ It is a Third-Order, Second-degree ODE.

Linear & Nonlinear Ordinary Differential Equation:

An equation in which the dependent variable and all its pertinent derivatives are of the first degree is called (Linear ODE), otherwise, the equation is said to be (Non-Linear).

Example:

(i) $\beta\frac{dy}{dx}+\frac{d^3y}{dx^3}=\rho e^{-x}$ It is linear, third-order differential equation.

(ii) $\frac{dy}{dx} + y\frac{d^3y}{dx^3} = e^{-x}$ It is non-linear, third-order differential equation.

A Solution to a differential Equation:

It is a function of independent variables that when replaced in the equation, produces an expression that can be reduced, through algebraic manipulation, to the form (0=0).

Example:

y=*sin*(*x*) is a solution to the equation $y'' + y = 0$, that –sin(x) +sin(x) =0

Boundary Conditions:

That are conditions provided at more than one value of the independent variables which evaluated at the boundaries of solution domain and its number is usually equal to the order of the ODE.

Non-Linear Boundary-Value Problem:

It is a non-linear differential equation which is determined by conditions given at two or more distinct points.

These problems can be solved numerically by iterative methods. In each iterative step a linear problem should be solved, that obtained by approximating non-linear equation to a system of linear equations.

A Finite Difference:

It is a mathematical expression of the form $f(x + b) - f(x + a)$. If a finite difference is divided by $b - a$, one gets a difference quotient. The approximation of derivatives by finite differences plays a central role in finite difference methods for the numerical solution of differential equations, especially boundary value problems.

Finite-Difference Method:

It is a general method used to solve boundary value ordinary differential equations, in which derivatives in equation are substituted by finite difference approximations to convert the ordinary differential equation into a system of equations with more than one unknown at different points (mesh points) at which the equation is valid and the resulting system can be solved using the suitable techniques for solving systems.

1.2 Deriving Finite-Difference Approximation:

To derive a finite difference approximation to any derivative say $u^{(n)}(\bar{x})$ based on a given set of points that there number must be greater than the derivative order. We will use Taylor series to derive an appropriate formula, and then use the method of undetermined coefficients.

Example (1.2): Find the finite difference approximation to $u''(\bar{x})$ based on $u(\bar{x})$, $u(\bar{x}-h)$ and $u(\bar{x}-2h)$, of the form:

$$u''(\bar{x}) = au(\bar{x}) + bu(\bar{x}-h) + cu(\bar{x}-2h) \qquad \dots (1.1)$$

Our aim is to determine the coefficients *a, b* and *c*, so the first step is:

Expanding $u(\bar{x}-h)$ and $u(\bar{x}-2h)$ in Taylor series:

$$bu(\bar{x}-h) = bu(\bar{x}) - bhu'(\bar{x}) + b\frac{h^2}{2}u''(\bar{x}) - b\frac{h^3}{6}u'''(\bar{x}) + \cdots \qquad \dots(1.2)$$

$$cu(\bar{x}-2h) = cu(\bar{x}) - 2chu'(\bar{x}) + 4c\frac{h^2}{2}u''(\bar{x}) - 8c\frac{h^3}{6}u'''(\bar{x}) + \cdots \dots(1.3)$$

By adding $au(\bar{x})$ to the equations (1.2) & (1.3) and collecting the terms:

$$au(\bar{x}) + bu(\bar{x} - h) + cu(\bar{x} - 2h) = (a + b + c)u(\bar{x}) - (b + 2c)hu'(\bar{x}) + (b + 4c)\frac{h^2}{2}u''(\bar{x}) - (b + 8c)\frac{h^3}{6}u'''(\bar{x}) + \cdots \quad \ldots(1.4)$$

The second step is finding the values of *a, b* and *c*:

We have now three unknowns and we get a system of three equations by putting the coefficients of any order except second-order –as we want u'' - to be zero:

$$\begin{aligned} a + b + c &= 0 \\ b + 2c &= 0 \\ b + 4c &= \frac{2}{h^2} \end{aligned} \quad \ldots (1.5)$$

By solving the above system using MATLAB 7.0 we get the coefficients:

$$a = \frac{1}{h^2}\ , b = -\frac{2}{h^2}\, , c = \frac{1}{h^2} \quad \ldots(1.6)$$

So the difference formula is:

$$u''(\bar{x}) \approx \frac{u(\bar{x}) - 2u(\bar{x} - h) + u(\bar{x} - 2h)}{h^2} + O(h) \quad \ldots(1.7)$$

In this thesis we construct an algorithm that finds finite difference approximations to any derivative using a set of $u(\bar{x} + ah)$ and another two algorithms one that uses a set of $u(\bar{x} + ah)$ and even known derivatives and the other uses a set of $u(\bar{x} + ah)$and any set of derivatives we want.

Algorithm (1.1): See MATLAB 7.0 program (p.183-184)

To find the finite-difference approximation for the derivative $u^{(o)}(x_i)$.

Step (1): Input the order of derivative (*o*), the number of $u(\bar{x} + ah)$ terms set (*N*) and the coefficients of *h* (*a*).

Step (2): Form (A) the power of h coefficients matrix with size $N \times N$ for the system (1.4) as:

(i) The first row is ones, (The first term coefficients in Eq.(1.4))

(ii) For i=2,3,…,N and j=1,2,3,…,N , set $A_{i,j}=a(j)^{(i-1)}$

(The rest terms coefficients in Eq. (1.4))

Step(3): Form the constant vector(b) for system (1.4), which are zeros except the terms when $i = o+1$ the derivative order so, For i=1 , 2, 3 ,. . . ,N

(i) If $i=o+1$ set b_i= factorial of the order (o).

(ii) else set $b_i=0$.

Step(4): Solve the linear system $Au = b$ using any Direct numerical method for solving the linear system(Gaussian Elimination, LU-Decomposition,…), to get the coefficients vector for the set of $u(\bar{x} + ah)$.

Step(5): Determine the order of (h) in the denominator for derivative approximation

(i) set h as unreal symbol

(ii) set $c= h^o$

Step(6): Output is the coefficients vector for $u(\bar{x} + ah)$set to approximate the derivative .$u^{(o)}(\bar{x}) \approx \frac{c_1u(\bar{x}+a_1h)+c_2u(\bar{x}+a_2h)+c_3u(\bar{x}+a_3h)+\cdots+c_Nu(\bar{x}+a_Nh)}{h^o}$

1.3 The Finite-Difference Approximation for Derivative Using Known Derivatives and Set of Points:

This method depends on the previous method with some modifications.

We use a set of known derivatives at endpoint and a set of points. Then we need to form linear system that can be solved by direct method.

Example (1.3): Derive the finite-difference approximation for third-order derivative y''' at i=1 using $u_0', u_0'', u_{i-1}, u_i, u_{i+1}, u_{i+2}$.

So this means that we use the first and second derivatives at x_i or $x_{i-1}, i = 0$

Now we rewrite the derivative in the form

$$u'''(x_1) = au(x_1) + bu(x_1 - h) + cu(x_1 + h) + du(x_1 + 2h) + ehu'(x_1 - h) + fh^2u''(x_1 - h) \quad \ldots(1.8)$$

Our aim is to find the coefficients *a, b*, c*, d, e and f* so the first step is:

Expanding $u(x_1 \pm h)$, $u(x_1 + 2h), u'(x_1 - h), u''(x_1 - h)$ using Taylor series:

$$bu(x_1 - h) = bu(x_1) - bhu'(x_1) + b\frac{h^2}{2}u''(x_1) - b\frac{h^3}{6}u'''(x_1) + b\frac{h^4}{24}u^{(iv)}(x_1) - b\frac{h^5}{120}u^{(v)}(x_1) + b\frac{h^6}{720}u^{(vi)}(x_1) + \ldots \quad \ldots(1.9)$$

$$cu(x_1 + h) = cu(x_1) + chu'(x_1) + c\frac{h^2}{2}u''(x_1) + c\frac{h^3}{6}u'''(x_1) + c\frac{h^4}{24}u^{(iv)}(x_1) + c\frac{h^5}{120}u^{(v)}(x_1) + c\frac{h^6}{720}u^{(vi)}(x_1) + \cdots \quad \ldots(1.10)$$

$$du(x_1 + 2h) = du(x_1) + 2dhu'(x_1) + 4d\frac{h^2}{2}u''(x_1) + 8d\frac{h^3}{6}u'''(x_1) + 16d\frac{h^4}{24}u^{(iv)}(x_1) + 32d\frac{h^5}{120}u^{(v)}(x_1) + 64c\frac{h^6}{720}u^{(vi)}(x_1) + \cdots \quad \ldots(1.11)$$

$ehu'(x_1 - h) = ehu'(x_1) - eh^2u''(x_1) + e\frac{h^3}{2}u'''(x_1) - e\frac{h^4}{6}u^{(iv)}(x_1) +$
$e\frac{h^5}{24}u^{(v)}(x_1) - e\frac{h^6}{120}u^{(vi)}(x_1) + \ \ldots$ ...(1.12)

$fh^2u''(x_1 - h) = fh^2u''(x_1) - fh^3u'''(x_1) + f\frac{h^4}{2}u^{(iv)}(x_1) - f\frac{h^5}{6}u^{(v)}(x_1) +$
$f\frac{h^6}{24}u^{(vi)}(x_1) + \ \ldots$ ...(1.13)

By adding $au(x_1)$ to the equations (1.9), (1.10), (1.11), (1.12) & (1.13) and collecting the coefficients for the same terms we get:

$au(x_1) + bu(x_1 - h) + cu(x_1 + h) + du(x_1 + 2h) + ehu'(x_1 - h) +$
$fh^2u''(x_1 - h) = (a + b + c + d)u(x_1) + (-b + c + 2d + e)hu'(x_1) +$
$(b + c + 4d - 2e + 2f)\frac{h^2}{2}u''(x_1) + (-b + c + 8d + 3e - 6f)\frac{h^3}{6}u'''(x_1) +$
$(b + c + 16d - 4e + 12f)\frac{h^4}{24}u^{(iv)}(x_1) + \cdots$ ...(1.14)

The second step is finding the values of *a, b, c, d, e, and f*:

As we have five unknowns, we get a system of five equations by putting the coefficients of any derivative except third-order –as we want u''' - to be zero:

$$\begin{gathered} a + b + c + d = 0 \\ -b + c + 2d + e = 0 \\ b + c + 4d - 2e + 2f = 0 \\ -b + c + 8d + 3e - 6f = \frac{3!}{h^3} \\ b + c + 16d - 4e + 12f = 0 \\ -b + c + 32d + 5e - 20f = 0 \end{gathered} \qquad \ldots(1.15)$$

By solving the above linear system using MATLAB 7.0 we get the coefficients: $a = \frac{12}{h^3}\ , b = \frac{39}{4h^3}\, , c = \frac{9}{4h^3}\, , d = 0\, , e = \frac{15}{2h^3}\, , f = \frac{3}{2h^3}$ ...(1.16)

So the difference formula is:

$$u'''(x_1) \approx \frac{48u(x_1)+39u(x_1-h)+9u(x_1+h)+30hu'(x_1-h)+6h^2u''(x_1-h)}{4h^3} - \frac{h^3}{60}y^{(iv)}(\xi_1) \quad ..(1.17)$$

but in this thesis we construct an algorithm that finds finite difference approximations to any derivative we want using a set of points and any set of derivatives, which we used many times to form a finite-difference formula for boundary-value problems at the end points where the centered formula has not any value.

The most usefulness for this program is helping to construct and solve large matrix to derive the suitable finite-difference formula at a point satisfying the boundary conditions of the problem with best global error value for method, which needs long time and hard hand work calculations that we may repeat many times until finding best local truncation error with second order convergence.

Algorithm (1.2): See the MATLAB 7.0 program (p.186-190)

To find the finite-difference approximation for the derivative $u^{(o)}(x_i)$ using a set of points and a set of derivatives.

Step(1): Input the order of derivative (o), the number of $u(x+ah)$ terms set (N) and the coefficients of h (a), the number of derivatives (M), the order of derivatives (v(m)), the coefficients of (h) in the derivatives (am).

Step(2): Form (A) the matrix of the h coefficients power with size $(N+M)\times(N+M)$ for the system (1.114) as:

(i) The first row is ones until j=N, and zero's if j>N

(The first term of equation coefficients in equation 1.14)

(ii) For $i=2,3,\dots,N+M$ and $j=1,2,3,\dots,N$, set $A_{i,j}=a_j^{(i-1)}$.
(The rest terms coefficients in equation (1.14)

For j>N

if $i=v_j+1$, $A_{i,j}$= factorial(v_j);

else $A_{i,j}= am_j^{(i-vj-1)}$*(factorial(i-1)/factorial(abs(i-v_j-1)));

Step(3): Form the constant vector(b) for system (1.14), which is zeros except the terms when $i = o+1$ the derivative order so, For i=1 , … ,$N+M$

(iii) if $i=o+1$ set b_i = factorial(o).

(iv) else set $b_i = 0$.

Step(4): Solve the linear system $Au = b$ using any Direct numerical method for solving the linear system(Gaussian Elimination, LU-Decomposition,…), to get the coefficients vector for the $u(x+ah)$ and $d^{(i)}$ $u(x+ah)$set.

Step(5): Determine the order of (h) in the denominator for derivative

(iii) set h as unreal symbol

(iv) set $c= h^o$

Step(6): Output is the coefficients vector for $u(x+ah)$set to approximate the derivative

$$u^{(o)}(x) \approx \frac{c_1u(x+a_1h)+c_2u(x+a_2h)+\cdots+c_Nu(x+a_Nh)+\sum_i^m h^{v(i)}c_iu^{(v(i)}(x+a_{m(i)}h)}{h^o}$$

1.4 Local Truncation Error (LTE):

It is defined by replacing w_i by the true solution $u(x_i)$ in finite formula, in general the true solution $u(x_i)$ will not satisfy this equation exactly, and the discrepancy is the LTE which is denoted by τ_i.

Suppose we want find the local truncation error for the finite difference formula for the second-order derivative $u''(x_i) \approx \frac{u(x_{i-2})-2u(x_{i-1})+u(x_i)}{h^2}$

$$\tau_i = \frac{1}{h^2}(w_{i-2} - 2w_{i-1} + w_i) - u''(x_i) \qquad \ldots(1.18)$$

$$\tau_i = u_i'' - hu_i''' + O(h) - u''(x_i) \qquad \ldots(1.19)$$

$$\tau_i = -hu_i''' + O(h) \qquad \ldots(1.20)$$

Although u_i''' is in general unknown, it is some fixed function independent of h and so $\tau_i = O(h)$ as $h \to 0$.

Algorithm (1.4): See the MATLAB 7.0 program (p.190-192)

To find the local truncation error for finite-difference approximation, we use the equation: $\tau_k = \sum_{i=1}^{N} \frac{c_i u^{(o+k)}(x+a_i h)}{(o+k)} h^{(o+k)}$ … (1.21)

Where k=1,2,3,.. until finding $\tau_k \neq 0$.

Step (1): Input the order of derivative, the number of *u(x+ah)* and its h coefficient.

Step (2): Find the finite-difference approximation vector (co) for the derivative using Algorithm (1.1).

Step (3): Set k=o+2, T=0

Step (4): While T=0 do Steps (5- 13)

Step (5): Find the vector of h to the power (o+k)

For i=1,2,3,…,N set $A1_{k,i}=a(i)^{(o+k)}$

Step (6): Find the nonzero truncation error, $T=A1_{k,i}*co$

Step (7): if T=0 set k=k+1

Step (8): if T≠0 do steps (9-12)

The following steps are used to prevent a very small error, caused by machine, as well as, the use of rational formatting while finding (co).

We find that T=0 using hand calculations, but when we use the program with MATLAB 7.0 we get very small value of $T = 1 \times 10^{-16}$, so we follow these steps, which depend on the fact that (if the nominator of a real fraction equals zero so the fraction will be zero).

Step (9): Separate the fractions vector (co) to two vectors nominators vector and denominators vector as [n,d]=rat(co)

[n,d]: it is a MATLAB 7.0 function that separate the fraction vector to two vectors. (n= nominators vector and d= denominators vector).

Step (10): To get the exact value for each part of fraction it should be rounded to the nearest integer (to prevent the rational formatting).

Step (11): As (co) has different fractions so we must uniform their denominators, which is the vector (d) from step (9):

Find the least common divisor for denominator vector (d), but there are no MATAB function that gets it directly, so we follow these steps:

(i) set N1=length of vector (d) , k1=1, m=maximum entry of (d), e=1

(ii) while e≠0 do

(iii) For i=1, 2, 3,…, N1 L_i=lcm(d_i ,m) lcm: MATLAB function that finds the least common multiple for two integers.

To test if denominator vector is uniform or not:

(iv) set G vector of the maximum value of (L) ,

(v) set e= maximum entry of vector |L-G|

If the maximum difference between two positive integers equals zero this means that the integers are equal, else they are not equal.

(vi) test if $e \neq 0$ set k=k+1 and repeat steps(ii-v)

(vii) if e=0 Stop and do the steps (viii-xi) so we uniform the denominator vector.

(viii) Now we must uniform the fractions by getting the number that multiplies (d) to be uniformed.

(ix) set m1=divide the entries of L on entries of d for i=1,2,3,…N1)

(x) set new $n^{(k)}$=multiply the entries of n by the entries of m1 for i=1,2,3,.. N1)

(xi) set T1= $A1_{k,i} * n^{(k)}$

Step (12): if T1 =0, set k=k+1

Step (13): if $T1 \neq 0$, Stop

tn=($A1_{k,i}$ *co)/factorial for (k-1)

Step (14): To determine the order of truncation error *O(h):*

(i) set (h) unreal symbol.

(ii) $c1 = h^{(k-o-1).}$

Step (15): Output the local truncation is (tn) with order (c1) .

The three previous algorithms are very important for our thesis that gets us the suitable form we need for derivative with suitable order of truncation error on short time.

For example, the hand calculations for deriving the finite-difference formula for a third-order derivative (y_i''') in example (1.3) need about 30 minutes, but using these algorithms we need 0.047 seconds.

1.5 Newton's Method for Non-Linear System:

This method can be summarized by this equation:

$$\boldsymbol{x}^k = \boldsymbol{x}^{k-1} - J^{-1}(\boldsymbol{x}^{k-1})F(\boldsymbol{x}^{k-1})$$

Where *F* is the non-linear system that we want to solve and must be written on the form *F(x) =**0***.

J is the Jacobin N×N matrix which is given as:

$$\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_N} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_N}{\partial x_1} & \frac{\partial f_N}{\partial x_2} & \cdots & \frac{\partial f_N}{\partial x_N} \end{bmatrix}$$

to solve the above equation we will select initial x^0 guess and generating for $k \geq 1$ then solve $Jy = -F$ to get the vector increment of y then

$$x^{(k)} = x^{(k-1)} + y^{(k-1)}$$

and so on until reaching convergence or exceeding the maximum number of iterations.

CHAPTER TWO

Third-Order Linear Boundary-Value Problem

Linear third-order boundary-value problems have many applications in physics, engineering, and chemistry, see [1], [5], &[8].

In this chapter our concern is approximating the solution for a third-order linear boundary-value problem by using finite-difference approximations with second order convergence and then we will use Richardson's Extrapolation methods to obtain a better approximation for the problem.

There are many cases for the linear third-order boundary-value problems. These have different forms of boundary conditions; there solution existence and uniqueness are discussed widely by several papers, see [1].

In this chapter we will discuss the numerical solution for four cases of the problem using finite-difference method.

2.1 Finite-Difference Method (2.1) for Linear Third-Order Boundary-Value Problem:

Consider the linear third-order boundary-value problem:

$$y'''(x) = p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \qquad a \le x \le b \quad \ldots(2.1)$$

with first especial case of three boundary conditions

$$y(a) = \alpha_0 \ , y'(a) = \alpha_1 \, , y(b) = \beta_0 \qquad \ldots(2.2)$$

First, we select an integer *N*>0 and divide the interval [*a,b*] into (*N*+1) equal subintervals whose endpoints are the mesh points $x_i = a + ih$ for *i*=0,1,2,…,*N+1*, where the step size *h=(b-a)/(N+1)*.

Second, we approximate all derivatives in (2.1) using finite-difference approximation on the interior mesh points.

$$y'(x_i) = \frac{y(x_{i+1}) - y(x_{i-1})}{2h} - \frac{h^2}{6} y'''(\xi_i) \qquad \ldots(2.3)$$

$$y''(x_i) = \frac{y(x_{i-1}) - 2y(x_i) + y(x_{i+1})}{h^2} - \frac{h^2}{12} y^{(iv)}(\eta_i) \qquad \ldots(2.4)$$

which are centered-difference formulas with second order convergence.

But we take special finite-difference approximation for the third-derivative, to be suitable to the problem boundary conditions and maintain the second order convergence.

$$y'''(x_i) = \frac{y(x_{i-3}) - 6y(x_{i-2}) + 12y(x_{i-1}) - 10y(x_i) + 3y(x_{i+1})}{2h^3} + \frac{h^2}{4} y^{(v)}(\zeta_i) \qquad \ldots(2.5)$$

Now the substituting of the equations (2.3), (2.4) and (2.5) in Eq.(2.1) gets the equation

$\frac{y(x_{i-3})-6y(x_{i-2})+12y(x_{i-1})-10y(x_i)+3y(x_{i+1})}{2h^3} = p_2(x_i)[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}] + p_1(x_i)[\frac{y(x_{i+1})-y(x_{i-1})}{2h}] + p_0(x_i)y(x_i) + r(x_i) - \frac{h^2}{12}[2p_1(x_i)y^{'''}(\xi_i) + p_2(x_i)y^{(iv)}(\eta_i) + 3y^{(v)}(\zeta_i)]$ …(2.6)

A Finite-Difference method with truncation error of order $O(h^2)$ results by using (2.6) together with the boundary conditions (2.2) to define

$$w_0 = \alpha_0\ , \qquad w_{N+1} = \beta_0, \qquad w_0^{'} = \alpha_1$$

and

$\frac{w_{i-3}-6w_{i-2}+12w_{i-1}-10w_i+3w_{i+1}}{2h^3} - p_2(x_i)\left[\frac{w_{i-1}-2w_i+w_{i+1}}{h^2}\right] - p_1(x_i)\left[\frac{w_{i+1}-w_{i-1}}{2h}\right] - p_0(x_i)w_i = r(x_i)$ …(2.7)

for each *i*= 3,4,5,…,*N* .

Now by multiplying Eq.(2.7) by $2h^3$ it can be written:

$w_{i-3} - 6w_{i-2} + [12 - 2hp_2(x_i) + h^2p_1(x_i)]w_{i-1} - [10 - 4hp_2(x_i) + 2h^3p_0(x_i)]w_i + [3 - 2hp_2(x_i) - h^2p_1(x_i)]w_{i+1} = 2h^3r(x_i)$ …(2.8)

for each *i*= 3,4,5,…,*N* .

but this formula is suitable for N-2 mesh points, so we need to derive special formula when *i*=1,2 using the boundary conditions

$$y_1^{'} = \frac{y_2-y_0}{2h} - \frac{h^2}{6}y^{'''}(\xi_1) \qquad \text{…(2.9)}$$

$$y_1^{''} = \frac{y_0-2y_1+y_2}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_1) \qquad \text{…(2.10)}$$

In order to maintain second order convergence, we derive new finite-difference approximation using the first derivative at *i*=0 and one more point in difference formula using special algorithm (1.2) which is given by

$$y_1''' = \frac{8y_0 - 9y_1 + y_3 + 6hy_0'}{3h^3} - \frac{3h^2}{20} y^{(v)}(\zeta_1) \qquad \ldots(2.11)$$

by substituting equations (2.9),(2.10)and (2.11) in Eq.(2.1) we get

$$\frac{8y_0 - 9y_1 + y_3 + 6hy_0'}{3h^3} = p_2(x_1)\left[\frac{y_0 - 2y_1 + y_2}{h^2}\right] + p_1(x_1)\left[\frac{y_2 - y_0}{2h}\right] + p_0(x_1)y_1 + r(x_1) - \frac{h^2}{60}[10p_1(x_1)y'''(\xi_1) + 5p_2(x_1)y^{(iv)}(\eta_1) - 9y^{(v)}(\zeta_1)] \qquad \ldots(2.12)$$

by omitting the error term and multiplying the previous equation by $6h^3$ we get the finite-difference formula at *i*=1

$$[16 - 6hp_2(x_1) + 3h^2p_1(x_1)]w_0 - [18 - 12hp_2(x_1) + 6h^3p_0(x_1)]w_1 - [6hp_2(x_1) + 3h^2p_1(x_1)]w_2 + 2w_3 + 12hw_0' = 6h^3r(x_1) \qquad \ldots(2.13)$$

To find the suitable finite-difference formula at *i*=2 we follow the same step as *i*=1, but without using more one point in deriving it

$$y_2' = \frac{y_3 - y_1}{2h} - \frac{h^2}{6} y'''(\xi_2) \qquad \ldots (2.14)$$

$$y_2'' = \frac{y_1 - 2y_2 + y_3}{h^2} - \frac{h^2}{12} y^{(iv)}(\eta_2) \qquad \ldots(2.15)$$

$$y_2''' = \frac{-14y_0 + 27y_1 - 18y_2 + 5y_3 - 6hy_0'}{3h^3} + \frac{3h^2}{20} y^{(v)}(\zeta_2) \qquad \ldots(2.16)$$

by substituting equations (2.14),(2.15)and (2.16) on Eq.(2.1) we get

$$\frac{-14y_0 + 27y_1 - 18y_2 + 5y_3 - 6hy_0'}{3h^3} = p_2(x_2)\left[\frac{y_1 - 2y_2 + y_3}{h^2}\right] + p_1(x_2)\left[\frac{y_3 - y_1}{2h}\right] + p_0(x_2)y_2 + r(x_2) - \frac{h^2}{60}[10p_1(x_2)y'''(\xi_2) + 5p_2(x_2)y^{(iv)}(\eta_2) + 9y^{(v)}(\zeta_2)] \qquad \ldots(2.17)$$

by omitting the error term and multiplying the previous equation by $6h^3$ we get the finite-difference formula at *i*=2

$$-28w_0 + [54 - 6hp_2(x_2) + 3h^2p_1(x_2)]w_1 - [36 - 12hp_2(x_2) + 6h^3p_0(x_2)]w_2 + [10 - 6hp_2(x_2) - 3h^2p_1(x_2)]w_3 - 12hw_0' = 6h^3r(x_2)$$

…(2.18)

Using equations (2.8), (2.13), and (2.18) together give a system with *N*×*N* nearly penta-diagonal matrix problem

$$Aw = \mathrm{c} \qquad \text{…(2.19)}$$

where

$$A = \begin{bmatrix} -(18+3F_2(x_1)) & -3F_3(x_1) & 2 & & & 0 \\ 54-3F_1(x_2) & -(36+3F_2(x_2)) & 10-3F_3(x_2) & & & \\ -6 & 12-F_1(x_3) & -(10+F_2(x_3)) & 3-F_3(x_3) & & \\ 1 & -6 & 12-F_1(x_4) & -(10+F_2(x_4)) & 3-F_3(x_4) & \\ & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & & 1 & -6 & 12-F_1(x_N) & -(10+F_2(x_N)) \end{bmatrix}_{N\times N}$$

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{N-1} \\ w_N \end{bmatrix}_{N\times 1}, \text{ and } c = \begin{bmatrix} 6h^3r(x_1) - \big(16+3F_1(x_1)\big)\alpha_0 - 12h\alpha_1 \\ 6h^3r(x_2) + 28\alpha_0 + 12h\alpha_1 \\ 2h^3r(x_3) - w_0 \\ 2h^3r(x_4) \\ \vdots \\ 2h^3r(x_{N-1}) \\ 2h^3r(x_N) - (3-F_3(x_N))\beta_0 \end{bmatrix}_{N\times 1}$$

$F_1(x_i) = 2hp_2(x_i) - h^2p_1(x_i)$, $F_2(x_i) = -4hp_2(x_i) + 2h^3p_0(x_i)$
$F_3(x_i) = 2hp_2(x_i) + h^2p_1(x_i)$, for i=1,2,3,…,N

Now, using LU-decomposition with MATLAB, we will solve the linear system (2.19), to reduce calculations at several values of step-size (h=0.1, 0.05, 0.025), then we will extrapolate the solution to get higher-order accuracy.

Algorithm (2.1): Linear Finite-Difference Method (2.1) for Solving Third-Order BVPS Case (I): See full MATLAM program (p.193-197)

To approximate the solution for the boundary-value problem

$$y''' = p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \qquad a \le x \le b$$

with boundary conditions $y(a) = \alpha_0,\ y(b) = \beta_0,\ y'(a) = \alpha_1$

Step(1):Input endpoints a, b; boundary conditions $\alpha_0, \beta_0, \alpha_1$

Step(2): For k=1,2,3,4 (To determine h=0.1,0.05,0.025,0,0125)

- set $N_k=(10*2^{(k-1)})-1$;
- $h=(b-a)/(N_k+1)$;
- do steps(2-9)

Step(3): For i=1,2,3, …,N_k set $x(i)=a+ih$, find $p_0(x_i)$, $p_1(x_i)$, $p_0(x_i)$, and $r(x_i)$. (The values of associated functions on Eq.(2.1))

Step(4): For $i=1,2,3,\dots,N_k$ set

- $F_1(i)= 2hp_2(i) - h^2p_1(i)$;
- $F_2(i)=-2h^3p_0(i) + 4hp_2(i)$;
- $F_3(i)= 2hp_2(i) + h^2p_1(i)$;

(The adding Eq.(2.3)&(2.4) finite-difference formula for y'',y'as F's)

Step(5): To determine the diagonals for matrix A in the system (2.19) ; For i=1,2,3,…,N_k set

(The main diagonal for A (d0))

d0=[-(18+3F_2(1)) -(36+3F_2(2)) -(10+F_2(3)) ….. –(10+F_2(N_k))];

(The first upper diagonal for A (dU1))

dU1=[$-3F_3(1)$ $(10-3F_3(2))$ $(3-F_3(3))$ ……. $(3-F_3(N_k-1))$];

(The second diagonal for A (dU2))

dU2=$[2\ \ 0\ \ \ldots\ \ 0]_{1\times(N_k-2)}$

(The first lower diagonal for A (dL1))

dL1=[$(54+3F_1(2))$ $(12+F_1(3))$ …… $(12+F_1(N_k)$] ;

(The second lower diagonal for A (dL2))

dL2= $[-6\ \ \ldots\ -6]_{1\times(N_k-1)}$

(The third lower diagonal for A (dL3))

dL3=$[1\ \ \ldots 1]_{1\times(N_k-2)}$

(The constant vector on (c) Eq.(2.19))

c=[$(12h^3r(1)-(16-3F_1(1))a0-6h\mathrm{a}1)$;$(12h^3r(2)+28a0+6h\mathrm{a}1)$;($2h^3r(3)-a0$); $2h^3r(4)$; …..;$2h^3r(N(k)-1)$; $2h^3r(N(k))-(3-F_3(N(k)))\beta_0$]

Step(6) Factorize A to two diagonals matrix: matrix L has main diagonal-ones entries- and other three lower ,and U has main diagonal and two upper

Step(7) For $i=1,2,3,\ldots,N_k$ Solve LY=d to find Y using forward substitution

Step(8) For $i=N,\ N_k-1\ ,\ldots,\ 2,\ 1$ Solve Uw=Y to find the approximation solution w using backward substitution

To write the w's values at $x_i=a+0.1*i$, for i=0,1,…,9

Step(9) Set N1=9

For $i=1,2,3,\ldots N_k$, for j= 0,2,4,8,…,N_k set $W1_{i,k}=w_j$;

Step(10) For $i=1,2,\ldots,N1$ set h=(b-a)/N1 ; x(i)=a+ih

Step(11) Extrapolate the solution at h= 0.1,0.05,0.025,0,0125 by set

Ext2(h=0.1)=(4*W1(h=0.05)-W1(h=0.1))/3;

Ext2(h=0.05)=(4*W1(h=0.025)-W1(h=0.05))/3;

Ext2(h=0.025)=(4*W1(h=0.0125)-W1(h=0.025))/3;

Ext3(h=0.1)=(16*Ext2(h=0.05)-Ext2(h=0.1))/15;

Ext3(h=0.05)=(16*Ext2(h=0.025)-Ext2(h=0.05))/15;

Ext4(h=0.1)=(64*Ext3(h=0.05)-Ext3(h=0.1))/63;

Step(12) Output [x' ,W1',Ext4] (The approximation solutions and Extrapolation)

Example 1: Consider the linear third-order boundary value problem

$y'''(x) = 2x^2y'' - 3xy' - 5x^2y + e^{2x}(3x^3 - x^2 - 5x - 4), 0 \leq x \leq 1$

with boundary conditions : $y(0) = 1, \quad y(1) = 0, \quad y'(0) = 1$

So $p_2(x) = 2x^2$, $p_1(x) = -3x$, $p_0(x) = -5x^2$, and

$r(x) = e^{2x}(3x^3 - x^2 - 5x - 4)$ on the interval $[0,1]$

Now by applying algorithm (2.1) to the example we get the following results that approximated solutions at several values of h as listed in Table (2.1)

Table 2.1

x_i	Exact Y	$w_i(h=0.1)$	$w_i(h=0.05)$	$w_i(h=0.025)$	$w_i(h=0.0125)$	Ext3(0.1)	Ext4(0.1)
0.1	1.09926248	1.09945718	1.09934530	1.09928723	1.09926910	1.09926520	1.09926269
0.2	1.19345976	1.19433658	1.19379324	1.19355538	1.19348492	1.19346702	1.19346036
0.3	1.27548316	1.27753316	1.27620571	1.27568575	1.27553614	1.27549571	1.27548435
0.4	1.33532456	1.33890771	1.33652472	1.33565662	1.33541109	1.33534304	1.33532645
0.5	1.35914091	1.36439876	1.36084026	1.35960702	1.35926211	1.35916540	1.35914355
0.6	1.32804677	1.33481404	1.33017748	1.32862762	1.32819757	1.32807627	1.32805003
0.7	1.21655999	1.22425193	1.21893383	1.21720414	1.21672703	1.21659201	1.21656358
0.8	0.99060648	0.99806306	0.99287133	0.99121882	0.99076513	0.99063646	0.99060989
0.9	0.60496475	0.61022938	0.60654294	0.60539014	0.60507488	0.60498532	0.60496710

The previous results reveal that as h is decreases the accuracy increases. The maximum error for each approximation is as listed in Table (2.2)

Table 2.2

$\|Y-W\|$	Maximum Error
$\|Y - w_i(h = 0.1)\|$	7.7×10^{-3}
$\|Y - w_i(h = 0.05)\|$	2.4×10^{-3}
$\|Y - w_i(h = 0.025)\|$	6.4415×10^{-4}
$\|Y - w_i(h = 0.0125)\|$	1.6704×10^{-4}
$\|Y - Ext3_i(h = 0.1)\|$	3.2016×10^{-5}
$\|Y - Ext4_i(h = 0.1)\|$	1.6704×10^{-6}

Table (2.2) values reveal that the best error is at the Fourth-Extrapolation when h=0.1.

2.2 Finite-Difference Method (2.2) for Linear Third-Order Boundary-Value Problem:

Consider the linear third-order boundary-value problem:

$$y'''(x) = p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \qquad a \le x \le b \qquad ..(2.20)$$

with second especial case of three boundary conditions

$$y(a) = \alpha_0 \ , \ y(b) = \beta_0 \ \ , y''(b) = \beta_2 \qquad ...(2.21)$$

To approximate the solution to this problem we will follow the same steps in the previous by dividing the interval [*a,b*] into N+1 equal subintervals whose endpoints $x_i=a+ih$,for $i=0,1,2,...,N+1$, then we substitute all derivatives by appropriate finite-difference formulas.

By using algorithm (1.1) we get a suitable finite-difference approximation for y'''

$$y'''(x_i) = \frac{-3y(x_{i-1})+10y(x_i)-12y(x_{i+1})+6y(x_{i+2})-y(x_{i+3})}{2h^3} + \frac{h^2}{4}y^{(v)}(\zeta_i) \quad ..(2.22)$$

The use of this formula maintains the convergence of this method and decreases the use of one more points at endpoints where their boundary conditions are not sufficient.

Now we use the equations (2.3), (2.4) centered-difference formula for $y', y''$and Eq.(2.22) and substituting in Eq.(2.20) to get the equation

$$\frac{-3y(x_{i-1})+10y(x_i)-12y(x_{i+1})+6y(x_{i+2})-y(x_{i+2})}{2h^3} =$$
$$p_2(x_i)\left[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}\right] + p_1(x_i)[\frac{y(x_{i+1})-y(x_{i-1})}{2h}] + p_0(x_i)y(x_i) +$$
$$r(x_i) - \frac{h^2}{12}[2p_1(x_i)y'''(\xi_i) + p_2(x_i)y^{(iv)}(\eta_i) + 3y^{(v)}(\zeta_i)] \qquad ...(2.23)$$

A Finite-Difference method with truncation error of order $O(h^2)$ results by using (2.23) together with the boundary conditions (2.21) to define

$$w_0 = \alpha_0 , \qquad w_{N+1} = \beta_0, \qquad w''_{N+1} = \beta_2$$

and

$$\frac{-3w(x_{i-1})+10w(x_i)-12w(x_{i+1})+6w(x_{i+2})-w(x_{i+2})}{2h^3} - p_2(x_i)\left[\frac{w_{i-1}-2w_i+w_{i+1}}{h^2}\right] - p_1(x_i)\left[\frac{w_{i+1}-w_{i-1}}{2h}\right] - p_0(x_i)w_i = r(x_i) \qquad \ldots (2.24)$$

for each i=1,2, 3,4,5,…,N-2 .

Now by multiplying Eq.(2.24) by $2h^3$ it can be written as :

$$-[3+2hp_2(x_i)-h^2p_1(x_i)]w_{i-1}+[10+4hp_2(x_i)-2h^3p_0(x_i)]w_i - [12+2hp_2(x_i)+h^2p_1(x_i)]w_{i+1}+6w_{i+2}-w_{i+3} = 2h^3r(x_i) \qquad \ldots(2.25)$$

but this equation is suitable for *N-2* mesh points that we must derive a special formula that maintains the second order convergence at *i=N-1,N*

Now we will derive the formula when *i=N-1* using boundary conditions

$$y'_{N-1} = \frac{y_N-y_{N-2}}{2h} - \frac{h^2}{6}y'''(\xi_{N-1}) \qquad \ldots(2.26)$$

$$y''_{N-1} = \frac{y_{N-2}-2y_{N-1}+y_N}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_{N-1}) \qquad \ldots(2.27)$$

$$y'''_{N-1} = \frac{-17y_{N-2}+57y_{N-1}-63y_N+23y_{N+1}-6h^2y''_{N+1}}{11h^3} + \frac{9h^2}{44}y^{(v)}(\zeta_{N-1}) \qquad \ldots(2.28)$$

By substituting equations (2.26), (2.27) and (2.28) in (2.20) we get

$$\frac{-17y_{N-2}+57y_{N-1}-63y_N+23y_{N+1}-6h^2y''_{N+1}}{11h^3} = p_2(x_{N-1})\left[\frac{y_{N-2}-2y_{N-1}+y_N}{h^2}\right] + p_1(x_{N-1})\left[\frac{y_N-y_{N-2}}{2h}\right] + p_0(x_{N-1})y_{N-1} + r(x_{N-1}) - \frac{h^2}{132}[22p_1(x_{N-1})y'''(\xi_{N-1}) + 11p_2(x_{N-1})y^{(iv)}(\eta_{N-1}) + 18y^{(v)}(\zeta_{N-1})] \qquad \ldots(2.29)$$

By omitting the error term and multiplying the previous equation by $22h^3$ we get the finite-difference formula at *i=N-1*

$$-[34 + 22hp_2(x_{N-1}) - 11h^2p_1(x_{N-1})]w_{N-2} + [114 + 44hp_2(x_{N-1}) - 22h^3p_0(x_{N-1})]w_{N-1} - [126 + 22hp_2(x_{N-1}) + 11h^2p_1(x_{N-1})]w_N + 46w_{N+1} - 12h^2w''_{N+1} = 22h^3r(x_{N-1}) \qquad \ldots(2.30)$$

Finally, to derive finite-difference formula at *i=N*, we use

$$y'_N = \frac{y_{N+1}-y_{N-1}}{2h} - \frac{h^2}{6}y'''(\xi_N) \qquad \ldots(2.31)$$

$$y''_N = \frac{y_{N-1}-2y_N+y_{N+1}}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_N) \qquad \ldots(2.32)$$

In order to maintain second order convergence, we derive new finite-difference approximation for third-order derivative using the second derivative at *i=N* and one more point using special algorithm (1.2), which is given by

$$y'''_N = \frac{-5y_{N-2}+9y_{N-1}-3y_N-y_{N+1}+6h^2y''_{N+1}}{11h^3} - \frac{9h^2}{44}y^{(v)}(\zeta_N) \qquad \ldots(2.33)$$

By substituting equations (2.31), (2.32) and (2.33) in Eq.(2.20) we get

$$\frac{-5y_{N-2}+9y_{N-1}-3y_N-y_{N+1}+6h^2y''_{N+1}}{11h^3} = p_2(x_N)\left[\frac{y_{N-1}-2y_N+y_{N+1}}{h^2}\right] + p_1(x_N)\left[\frac{y_{N+1}-y_{N-1}}{2h}\right] + p_0(x_N)y_N + r(x_N) - \frac{h^2}{132}[22p_1(x_1)y'''(\xi_1) + 11p_2(x_1)y^{(iv)}(\eta_1) - 18y^{(v)}(\zeta_1)] \qquad \ldots(2.34)$$

By omitting the error term and multiplying the previous equation by $22h^3$ we get the finite-difference formula at *i=N*

$$-10w_{N-2} + [18 - 22hp_2(x_N) + 11h^2p_1(x_N)]w_{N-1} - [6 - 44hp_2(x_N) + h^3p_0(x_N)]w_N - [2 + 22hp_2(x_N) + 11h^2p_1(x_N)]w_{N+1} + 6h^2w''_{N+1} = 22h^3r(x_N) \qquad \ldots(2.35)$$

Using equations (2.25), (2.30), and (2.35) together give a system with $N \times N$ nearly penta-diagonal matrix problem

$$Aw = c \qquad \ldots(2.36)$$

Where

A=

$$\begin{bmatrix} 10 - F_2(x_1) & -(12 + F_3(x_1)) & 6 & -1 & & 0 \\ -(3 + F_1(x_2)) & 10 - F_2(x_2) & -(12 + 3F_3(x_2)) & 6 & -1 & \\ & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & & -(3 + F_1(x_{N-2})) & 10 - F_2(x_{N-2}) & -(12 + 3F_3(x_{N-2})) \\ & & & -(34 + 11F_1(x_{N-1})) & 126 - 11F_2(x_{N-1}) & -(126 + 11F_3(x_{N-1})) \\ 0 & & & -10 & 18 - 11F_1(x_N) & -(6 + 11F_2(x_N)) \end{bmatrix}_{N \times N}$$

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{N-1} \\ w_N \end{bmatrix}_{N \times 1} \quad ; \quad c = \begin{bmatrix} (3 + F_1(x_1))\alpha_0 + 2h^3 r(x_1) \\ 2h^3 r(x_2) \\ \vdots \\ \beta_0 + 2h^3 r(x_{N-2}) \\ -46\beta_0 + 12h^2\beta_2 + 22h^3 r(x_{N-1}) \\ (2 + 11F_3(x_N))\beta_0 - 12h^2\beta_2 + 22h^3 r(x_N) \end{bmatrix}_{N \times 1}$$

$$F_1(x_i) = 2hp_2(x_i) - h^2 p_1(x_i) \ ,$$

$$F_2(x_i) = -4hp_2(x_i) + 2h^3 p_0(x_i)$$

$$F_3(x_i) = 2hp_2(x_i) + h^2 p_1(x_i), \text{ for } i=1,2,3,\ldots,N$$

Now, using LU-decomposition with MATLAB, we will solve the linear system (2.36), to reduce calculations at several values of step-size (h=0.1, 0.05, 0.025), then we will extrapolate the solution to get higher-order accuracy.

Algorithm (2.2): Linear Finite-Difference Method (2.2) for Solving Third-Order BVPS Case (II): See full MATLAM program (p.197-202)

To approximate the solution for the boundary-value problem

$y^{(3)} = p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \qquad a \le x \le b$

with boundary conditions $y(a) = \alpha_0,\ y(b) = \beta_0,\ y''(b) = \beta_2$

We use the same steps in algorithm (2.1) except modification in steps (5) &(6) as:

Step(5): To determine the diagonals for matrix *A* in the system (2.36) ;

For i=1,2,3,…,N_k set (The main diagonal for A (d0))

d0=[10-F_2(1) 10-F_2(2) ….. (114-11F_2(N-1)) –(6+11F_2(N_k))];

(The first upper diagonal for A (dU1))

dU1=[-(12+F_3(1)) -(12+F_3(2)) ……. -(126+11F_3(N_k-1))];

(The second upper diagonal for A (dU2)):dU2=$[6 \ldots 6]_{1\times(N_k-2)}$

(The third upper diagonal for A (dU3)):dU3=$[-1 \ \ldots\ -1]_{1\times(N_k-3)}$;

(The first lower diagonal for A (dL1))

dL1=[(54+3F_1(2)) (12+F_1(3)) …… (12+F_1(N_k)] ;

(The second lower diagonal for A (dL2)) : dL2= $[\,0\ \ldots 0\ -10]_{1\times(N_k-2)}$

(The constant vector on (c) Eq.(2.36))

c=[$(2h^3r(1) + \big(3 + F_1(1)\big)a0)$; $2h^3r(3)$;…$2h^3r(N_k - 2) + b_0$;

$(-46b_0 + 12h^2b_2 + 22\,h^3r(N_k-1)\,;\,(2+11F_3(N_k))b_0 - 12h^2b_2 + 22r(N_k)]$

Step(6): Factorize A into LU, where L is lower tridiagonal matrix with unit main diagonal and U is upper fourth-diagonal.

Then we complete the other steps in algorithm to get the results.

Example 2: Consider the linear third-order boundary value problem

$$y'''(x) = 2x^2y'' - 3xy' - 5x^2y + e^{2x}(3x^3 - x^2 - 5x - 4), 0 \le x \le 1$$

with boundary conditions : $y(0) = 1$, $y(1) = 0$, $y''(1) = -4e^2$

So $p_2(x) = 2x^2$, $p_1(x) = -3x$, $p_0(x) = -5x^2$, and

$r(x) = e^{2x}(3x^3 - x^2 - 5x - 4)$ on interval $[0,1]$

Now by applying algorithm (2.2) to the example we get the following results that approximated solutions at several values of h as listed in Table (2.3)

Table 2.3

x_i	Exact Y	w_i(h=0.1)	w_i(h=0.05)	w_i(h=0.025)	w_i(h=0.0125)	Ext3(0.1)	Ext4(0.1)
0.1	1.09926248	1.09473860	1.09670179	1.09843355	1.09903139	1.09912112	1.09924730
0.2	1.19345976	1.18634766	1.18908380	1.19202138	1.19305661	1.19320088	1.19343204
0.3	1.27548316	1.26750392	1.26999876	1.27364690	1.27496522	1.27513179	1.27544567
0.4	1.33532456	1.32792833	1.32939255	1.33329311	1.33474718	1.33490747	1.33528027
0.5	1.35914091	1.35342815	1.35336365	1.35710591	1.35855714	1.35868741	1.35909306
0.6	1.32804677	1.32466256	1.32294828	1.32618503	1.32750650	1.32758974	1.32799893
0.7	1.21655999	1.21556413	1.21255574	1.21502730	1.21610862	1.21613770	1.21651617
0.8	0.99060648	0.99134977	0.98795838	0.98952776	0.99028264	0.99026575	0.99057126
0.9	0.60496475	0.60609462	0.60372300	0.60441868	0.60479665	0.60476512	0.60494357

The previous results reveal that as h decreases the accuracy increases. The maximum error for each approximation is as listed in Table (2.4)

Table 2.4

$\lvert Y-W\rvert$	**Maximum Error**
$\lvert Y-w_i(h=0.1)\rvert$	7.9792×10^{-3}
$\lvert Y-w_i(h=0.05)\rvert$	5.9320×10^{-3}
$\lvert Y-w_i(h=0.025)\rvert$	2.0350×10^{-3}
$\lvert Y-w_i(h=0.0125)\rvert$	5.8377×10^{-4}
$\lvert Y-Ext3_i(h=0.1)\rvert$	1.0003×10^{-4}
$\lvert Y-Ext4_i(h=0.1)\rvert$	4.7850×10^{-5}

But this table reveals that the results for method (2.1) is better than method (2.2) which means, that the derivatives order in boundary conditions have a control on the method accuracy.

2.3 Finite-Difference Method (2.3) for Linear Third-Order Boundary-Value Problem:

Consider the linear third-order boundary-value problem:

$$y'''(x) = p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \qquad a \le x \le b \quad ...(2.37)$$

with third especial case of three boundary conditions

$$y(a) = \alpha_0\ ,\ y'(a) = \alpha_1\ \ ,y''(b) = \beta_2 \qquad \dots(2.38)$$

To approximate the solution to this problem we will follow the same steps in the previous section by dividing the interval $[a,b]$ into N+1 equal subintervals whose endpoints $x_i=a+ih$,for $i=0,1,2,\dots,N+1$. Then we substitute all derivatives by appropriate finite-difference formulas.

By using algorithm (1.1) we get a suitable finite-difference approximation for y'''

$$y'''(x_i) = \frac{y(x_{i-3})-6y(x_{i-2})+12y(x_{i-1})-10y(x_i)+3y(x_{i+1})}{2h^3} - \frac{h^2}{4}y^{(v)}(\zeta_i) \quad \ldots(2.39)$$

Now we use equations (2.3), (2.4) the centered-difference formula for $y', y''$and Eq.(2.39), and substituting in Eq.(2.37) to get the equation

$$\frac{y(x_{i-3})-6y(x_{i-2})+12y(x_{i-1})-10y(x_i)+3y(x_{i+1})}{2h^3} = p_2(x_i)\left[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}\right] + p_1(x_i)\left[\frac{y(x_{i+1})-y(x_{i+1})}{2h}\right] + p_0(x_i)y(x_i) + r(x_i) - \frac{h^2}{12}[2p_1(x_i)y^{(3)}(\xi_i) + p_2(x_i)y^{(iv)}(\eta_i) - 3y^{(v)}(\zeta_i)] \quad \ldots(2.40)$$

A Finite-Difference method with truncation error of order $O(h^2)$ results by using (2.40) together with the boundary conditions (2.38) to define

$$w_0 = \alpha_0\,, \qquad w_0' = \alpha_1, \qquad w_{N+1}'' = \beta_2$$

and

$$\frac{w_{i-3} - 6w_{i-2} + 12w_{i-1} - 10w_i + 3w_{i+1}}{2h^3} - p_2(x_i)\left[\frac{w_{i-1} - 2w_i + w_{i+1}}{h^2}\right] - p_1(x_i)\left[\frac{w_{i+1}-w_{i-1}}{2h}\right] - p_0(x_i)w_i = r(x_i) \quad \ldots(2.41)$$

for each *i=3,4,5,…,N-1*

Now by multiplying Eq.(2.41) by $2h^3$ we can rewrite it as:

$$w_{i-3} - 6w_{i-2} + [12 - 2hp_2(x_i) + h^2p_1(x_i)]w_{i-1} - [10 - 4hp_2(x_i) + 2h^3p_0(x_i)]w_i + [3 - 2hp_2(x_i) - h^2p_1(x_i)]w_{i+1} = 2h^3r(x_i) \quad \ldots(2.42)$$

But this formula is suitable for *N-3* mesh points, so we need to derive special formula when *i*=1,2,*N* using the boundary conditions, that maintains the second order convergence for the method.

Now we will derive finite-difference formula at *i=1*

$$y_1' = \frac{y_2 - y_0}{2h} - \frac{h^2}{6} y'''(\xi_1) \qquad \ldots(2.43)$$

$$y_1'' = \frac{y_0 - 2y_1 + y_2}{h^2} - \frac{h^2}{12} y^{(iv)}(\eta_1) \qquad \ldots(2.44)$$

In order to maintain second order convergence, we derive new finite-difference approximation for third-order derivative by using the first derivative at *i*=0 and one more point in algorithm (2.1), as following

$$y_1''' = \frac{8y_0 - 9y_1 + y_3 + 6hy_0'}{3h^3} - \frac{3h^2}{20} y^{(v)}(\zeta_1) \qquad \ldots(2.45)$$

By substituting equations (2.43), (2.44) and (2.45) in Eq.(2.37) we get

$$\frac{8y_0 - 9y_1 + y_3 + 6hy_0'}{3h^3} = p_2(x_1)\left[\frac{y_0 - 2y_1 + y_2}{h^2}\right] + p_1(x_1)\left[\frac{y_2 - y_0}{2h}\right] + p_0(x_1)y_1 + r(x_1) - \frac{h^2}{60}[10p_1(x_1)y'''(\xi_1) + 5p_2(x_1)y^{(iv)}(\eta_1) - 9y^{(v)}(\zeta_1)] \qquad \ldots(2.46)$$

By omitting the error term and multiplying the previous equation by $6h^3$ we get the finite-difference formula at *i*=1

$$[16 - 6hp_2(x_1) + 3h^2p_1(x_1)]w_0 - [18 - 12hp_2(x_1) + 6h^3p_0(x_1)]w_1 - [6hp_2(x_1) + 3h^2p_1(x_1)]w_2 + 2w_3 + 12hw_0' = 6h^3r(x_1) \qquad \ldots(2.47)$$

To find the suitable finite-difference formula at *i*=2, we follow the same step as in *i*=1, but without using more one point in deriving it

$$y_2' = \frac{y_3 - y_1}{2h} - \frac{h^2}{6} y'''(\xi_2) \qquad \ldots(2.48)$$

$$y_2'' = \frac{y_1 - 2y_2 + y_3}{h^2} - \frac{h^2}{12} y^{(iv)}(\eta_2) \qquad \ldots(2.49)$$

$$y_2''' = \frac{-14y_0 + 27y_1 - 18y_2 + 5y_3 - 6hy_0'}{3h^3} + \frac{3h^2}{20} y^{(v)}(\zeta_2) \qquad \ldots(2.50)$$

By substituting equations (2.48), (2.49) and (2.50) in Eq.(2.37) we get

$\frac{-14y_0+27y_1-18y_2+5y_3-6hy_0'}{3h^3} = p_2(x_2)\left[\frac{y_1-2y_2+y_3}{h^2}\right] + p_1(x_2)\left[\frac{y_3-y_1}{2h}\right] +$
$p_0(x_2)y_2 + r(x_2) - \frac{h^2}{60}[10p_1(x_2)y'''(\xi_2) + 5p_2(x_2)y^{(iv)}(\eta_2) + 9y^{(v)}(\zeta_2)]$

…(2.51)

By omitting the error term and multiplying the previous equation by $6h^3$ we get the finite-difference formula at *i*=2

$-28w_0 + [54 - 6hp_2(x_2) + 3h^2p_1(x_2)]w_1 - [36 - 12hp_2(x_2) + 6h^3p_0(x_2)]w_2 +$
$[10 - 6hp_2(x_2) - 3h^2p_1(x_2)]w_3 - 12hw_0' = 6h^3r(x_2)$ …(2.52)

Now we will find suitable finite-difference formula at *i*=*N* which maintains second order convergence and specify the boundary conditions

$$y_N' = \frac{y_{N-2}-4y_{N-1}+3y_N}{2h} + \frac{h^2}{3}y'''(\xi_N) \quad \text{…(2.53)}$$

$$y_N'' = \frac{y_{N-2}-2y_{N-1}+y_N+h^2y_{N+1}''}{2h^2} - \frac{13h^2}{24}y^{(iv)}(\eta_N) \quad \text{…(2.54)}$$

$$y_N''' = \frac{y_{N-3}-21y_{N-2}+39y_{N-1}-19y_N+18h^2y_{N+1}''}{35h^3} - \frac{5h^2}{28}y^{(v)}(\zeta_N) \quad \text{…(2.55)}$$

By substituting equations (2.53), (2.54) and (2.55) in Eq.(2.37) we get

$\frac{y_{N-3}-21y_{N-2}+39y_{N-1}-19y_N+18h^2y_{N+1}''}{35h^3} = p_2(x_N)\left[\frac{y_{N-2}-2y_{N-1}+y_N+h^2y_{N+1}''}{2h^2}\right] +$
$p_1(x_N)\left[\frac{y_{N-2}-4y_{N-1}+3y_N}{2h}\right] + p_0(x_N)y_N + r(x_N) + \frac{h^2}{168}[56p_1(x_N)y^{(3)}(\xi_N) -$
$91p_2(x_N)y^{(iv)}(\eta_N) + 30y^{(v)}(\zeta_N)]$ …(2.56)

By omitting the error term and multiplying Eq.(2.56) by $70h^3$we get the finite-difference formula at *i*=N

$2w_{N-3} - [42 + 35hp_2(x_N) + 35h^2p_1(x_N)]w_{N-2} + [78 + 70hp_2(x_N) +$
$140h^2p_1(x_N)]w_{N-1} - [38 + 35hp_2(x_N) + 105h^2p_1(x_N) + 70h^3p_0(x_N)]w_N +$
$[36 - 35hp_2(x_N)]h^2w_{N+1}'' = 70h^3r(x_N)$ …(2.57)

Finally, using equations (2.42), (2.47),(2.52), and (2.57) together give a linear system with *N*×*N* nearly penta-diagonal matrix problem

$$Aw = \mathrm{c} \qquad \ldots(2.58)$$

Where

A=

$$\begin{bmatrix} -(18+3F_2(x_1)) & -3F_3(x_1) & 2 & & & 0 \\ 54-3F_1(x_2) & -(36+3F_2(x_2)) & 10-3F_3(x_2) & & & \\ -6 & 12-F_1(x_3) & -(10+F_2(x_3)) & 3-F_3(x_3) & & \\ 1 & -6 & 12-F_1(x_4) & -(10+F_2(x_4)) & 3-F_3(x_4) & \\ & \ddots & \ddots & \ddots & \ddots & \ddots \\ & 1 & -6 & 12-F_1(x_N) & -(10+F_2(x_{N-1})) & 3-F_3(x_{N-1}) \\ 0 & & 2 & -(42+F_4(x_N)) & 78-F_5(x_N) & -(38+F_6(x_N)) \end{bmatrix}_{N\times N}$$

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{N-1} \\ w_N \end{bmatrix}_{N\times 1} ; c = \begin{bmatrix} 6h^3r(x_1) - \big(16+3F_1(x_1)\big)\alpha_0 - 12h\alpha_1 \\ 6h^3r(x_2) + 28\alpha_0 + 12h\alpha_1 \\ 2h^3r(x_3) - \alpha_0 \\ 2h^3r(x_4) \\ \vdots \\ 2h^3r(x_{N-1}) \\ 70h^3r(x_N) - (36-35hp_2(x_N))h^2\beta_2 \end{bmatrix}_{N\times 1}$$

$$F_1(x_i) = 2hp_2(x_i) - h^2p_1(x_i)\ ,$$

$$F_2(x_i) = -4hp_2(x_i) + 2h^3p_0(x_i)$$

$$F_3(x_i) = 2hp_2(x_i) + h^2p_1(x_i), \text{ for } i=1,2,3,\ldots,N\text{-}1$$

$$F_4(x_N) = 35hp_2(x_N) + 35h^2p_1(x_N)\ ,$$

$$F_5(x_N) = -70hp_2(x_N) - 140h^2p_1(x_N)$$

$$F_6(x_N) = 35hp_2(x_N) + 105h^2p_1(x_N) + 70h^3p_0(x_N)$$

Now, using LU-decomposition with MATLAB, we will solve the linear system (2.19), to reduce calculations at several values of step-size (h=0.1, 0.05, 0.025), then we extrapolate the solution to get higher-order accuracy.

Algorithm (2.3): Linear Finite-Difference Method (2.3) for Solving Third-Order BVPS Case (III): See full MATLAM program (p.202-207)

To approximate the solution for the boundary-value problem

$$y^{(3)} = p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \qquad a \le x \le b$$

with boundary conditions $y(a) = \alpha_0$, $y'(a) = \alpha_1$, $y''(b) = \beta_2$

This algorithm is the same as algorithm (2.1) with some modifications in steps (4and 5) as:

Step(4): For $i=1,2,3,\ldots,N_k$ set

- $F_1(i)= 2hp_2(i) - h^2p_1(i)$;
- $F_2(i)=-2h^3p_0(i) + 4hp_2(i)$;
- $F_3(i)= 2hp_2(i) + h^2p_1(i)$
- $F_4(x_N) = 35hp_2(x_N) + 35h^2p_1(x_N)$,
- $F_5(x_N) = -70hp_2(x_N) - 140h^2p_1(x_N)$
- $F_6(x_N) = 35hp_2(x_N) + 105h^2p_1(x_N) + 70h^3p_0(x_N)$

(The adding Eq.(2.3) &(2.4) finite-difference formula for y'', y' as F's)

Step(5): To determine the diagonals for matrix A in the system (2.58);

For i=1, 2, 3…, N_k set (The main diagonal for A (d0))

d0= [-(18+3F_2(1)) -(36+3F_2(2)) -(10+F_2(3)) ….. –(39+F_6(N_k))];

(The first upper diagonal for A (dU1))

dU1=[-3F_3(1) (10-3F_3(2)) (3-F_3(3)) ……. (3-F_3(N_k-1))];

(The second diagonal for A (dU2))

dU2=$[2\ 0\ \dots\ 0]_{1\times(N_k-1)}$

(The first lower diagonal for A (dL1))

dL1=[(54+3F_1(2)) (12+F_1(3)) …… (78- F_5(N_k-1)] ;

(The second lower diagonal for A (dL2))

dL2= $[-6 \dots -6\ \ -42-F4(N)\ ]_{1\times(N_k-2)}$

(The third lower diagonal for A (dL3))

dL3=$[1 \dots 1\ \ 2]_{1\times(N_k-3)}$;

(The constant vector on (c) Eq.(2.58))

c=[$(12h^3r(1)-\left(16-3F_1(1)\right)a0-6ha1)$;$(12h^3r(2)+28a0+6ha1)$;($2h^3r(3)-a0$); $2h^3r(4)$; …..;$2h^3r(N_k-1)$; $70h^3r(x_N)-(36-35hp_2(x_N))h^2\beta_2$]

then we continue the algorithm to get the approximation vector w

Example 3: Consider the linear third-order boundary value problem

$y'''(x)=2x^2y''-3xy'-5x^2y+e^{2x}(3x^3-x^2-5x-4), 0\le x\le 1$

with boundary conditions : $y(0)=1,\quad y'(0)=1,\quad y''(1)=-4e^2$

So $p_2(x)=2x^2$, $p_1(x)=-3x$, $p_0(x)=-5x^2$, and

$r(x)=e^{2x}(3x^3-x^2-5x-4)$ on interval [*0,1*]

Now by applying algorithm (2.3) to the example we get results, which approximated the solutions for the problem at several values of h as listed in Table (2.5).

Table 2.5

x_i	Exact Y	w_i(h=0.1)	w_i(h=3.125^{-3})	w_i(h=1.5625^{-3})	w_i(h=7.8125^{-4})
0.1	**1.09926248**	**1.05859890**	**1.09883598**	**1.09915376**	**1.09923508**
0.2	**1.19345976**	**1.03091315**	**1.19175407**	**1.19302494**	**1.19335018**
0.3	**1.27548316**	**0.91005993**	**1.27164775**	**1.27450542**	**1.27523677**
0.4	**1.33532456**	**0.68658141**	**1.32851532**	**1.33358872**	**1.33488713**
0.5	**1.35914091**	**0.34789500**	**1.34852799**	**1.35643543**	**1.35845914**
0.6	**1.32804677**	**-0.12237751**	**1.31282786**	**1.32416712**	**1.32706911**
0.7	**1.21655999**	**-0.74516754**	**1.19598188**	**1.21131417**	**1.21523806**
0.8	**0.99060648**	**-1.54676537**	**0.96399897**	**0.98382364**	**0.98889724**
0.9	**0.60496475**	**-2.55960248**	**0.57179284**	**0.59650850**	**0.60283381**

We note from the results that the first approximation has bad values, so we decrease h values to approach more realistic approximations, and then we use Extrapolation method at several values of h. In this table we choose the first approximations that their maximum error less than 10^{-2}.

Table 2.6

X_i	Exact Y	Ext2 $h = 6.25^{-3}$	Ext3 $h = 1.25^{-2}$	Ext4 $h = 0.05$	Ext5 h =0.1	Ext6 h =0.1	Ext7 h =0.1
0.1	**1.09926248**	**1.09897925**	**1.09925114**	**1.09913578**	**1.09914145**	**1.09925393**	**1.09926159**
0.2	**1.19345976**	**1.19232705**	**1.19341438**	**1.19295305**	**1.19297575**	**1.19342555**	**1.19345619**
0.3	**1.27548316**	**1.27293622**	**1.27538113**	**1.27434382**	**1.27439485**	**1.27540624**	**1.27547513**
0.4	**1.33532456**	**1.33080297**	**1.33514343**	**1.33330191**	**1.33339250**	**1.33518800**	**1.33531030**
0.5	**1.35914091**	**1.35209385**	**1.35885863**	**1.35598857**	**1.35612976**	**1.35892809**	**1.35911869**
0.6	**1.32804677**	**1.31794180**	**1.32764200**	**1.32352659**	**1.32372904**	**1.32774160**	**1.32801490**
0.7	**1.21655999**	**1.20289742**	**1.21601272**	**1.21044848**	**1.21072220**	**1.21614739**	**1.21651691**
0.8	**0.99060648**	**0.97294186**	**0.98989892**	**0.98270489**	**0.98305880**	**0.99007304**	**0.99055078**
0.9	**0.60496475**	**0.58294356**	**0.60408271**	**0.59511459**	**0.59555579**	**0.60429977**	**0.60489531**

The Table (2.6) lists the first good approximations for the problem which reveal using Richardson's Extrapolation with maximum error less than 1×10^{-2} for second, third, fourth, fifth, sixth and seventh Extrapolation.

The previous results reveal that as h decreases the accuracy increases.

Now the maximum error for each approximation is as listed in Table(2.6)

Table 2.7

$\|Y-W\|$	**Maximum Error**
$\|Y-w_i(h=0.1)\|$	3.1664
$\|Y-w_i(h=3.125^{-3})\|$	3.3172×10^{-2}
$\|Y-w_i(h=1.6525^{-3})\|$	8.4562×10^{-3}
$\|Y-w_i(h=7.8125^{-4})\|$	2.1309×10^{-3}
$\|Y-Ext2_i(h=6.25^{-3})\|$	2.2032×10^{-3}
$\|Y-Ext3_i(h=1.25^{-2})\|$	8.8203×10^{-4}
$\|Y-Ext4_i(h=0.05)\|$	9.8502×10^{-3}
$\|Y-Ext5_i(h=0.1)\|$	9.4090×10^{-3}
$\|Y-Ext6_i(h=0.1)\|$	6.6498×10^{-4}
$\|Y-Ext7_i(h=0.1)\|$	6.9438×10^{-5}

The results in this method are worse than the previous methods because of β_0 absence, which confirm our conclusion.

2.4 Finite-Difference Method (2.4) for Linear Third-Order Boundary-Value Problem:

Consider the linear third-order boundary-value problem:

$$y'''(x) = p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \qquad a \le x \le b \qquad .. (2.59)$$

with third especial case of three boundary conditions

$$y'(a) = \alpha_1 \ , \ y(b) = \beta_0 \ \ , y'(b) = \beta_1 \qquad \ldots(2.60)$$

To approximate the solution to this problem we will follow the same steps in the previous sections by dividing the interval [a,b] into N+1 equal subintervals whose endpoints $x_i=a+ih$,for $i=0,1,2,\ldots,N+1$, then we substitute all derivatives by appropriate finite-difference formulas.

By using algorithm (1.1) we get a suitable finite-difference approximation for y'''

$$y'''(x_i) = \frac{-3y(x_{i-1})+10y(x_i)-12y(x_{i+1})+6y(x_{i+2})-y(x_{i+3})}{2h^3} + \frac{h^2}{4}y^{(v)}(\zeta_i) \quad \ldots(2.61)$$

Now we use the equations (2.3), (2.4) the centered-difference formula for $y', y''$and Eq.(2.61) and substituting in Eq.(2.59) to get the equation

$$\frac{-3y(x_{i-1})+10y(x_i)-12y(x_{i+1})+6y(x_{i+2})-y(x_{i+3})}{2h^3} =$$
$$p_2(x_i)\left[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}\right] + p_1(x_i)\left[\frac{y(x_{i+1})-y(x_{i-1})}{2h}\right] + p_0(x_i)y(x_i) +$$
$$r(x_i) - \frac{h^2}{12}[2p_1(x_i)y'''(\xi_i) - p_2(x_i)y^{(iv)}(\eta_i) - 3y^{(v)}(\zeta_i)] \qquad \ldots(2.62)$$

A Finite-Difference method with truncation error of order $O(h^2)$ results by using (2.62) together with the boundary conditions (2.60) to define

$$w_0' = \alpha_1, \qquad w_{N+1} = \beta_0\,, \qquad w_{N+1}' = \beta_1$$

and

$$\frac{-3w_{i-1} + 10w_i - 12w_{i+1} + 6w_{i+2} - w_{i+3}}{2h^3} - p_2(x_i)\left[\frac{w_{i-1} - 2w_i + w_{i+1}}{h^2}\right]$$
$$- p_1(x_i)\left[\frac{w_{i+1} - w_{i-1}}{2h}\right] - p_0(x_i)w_i\, r(x_i) \qquad \ldots(2.63)$$

for each $i=2,3,4,\ldots N-2$

Now by multiplying Eq.(2.63) by $2h^3$ we rewrite it in the form as

$$-(3 + 2hp_2(x_i) - h^2p_1(x_i))w_{i-1} + (10 + 4hp_2(x_i) - 2h^3p_0(x_i))w_i -$$
$$(12 + 2hp_2(x_i) + h^2p_1(x_i))w_{i+1} + 6w_{i+2} - w_{i+3} = 2h^3r(x_i) \quad \ldots(2.64)$$

but this formula is suitable for *N-3* mesh points, that it not suitable for *i=1,2 and N.*

We must derive formulas that are suitable for the three points and maintain the second order convergence using boundary conditions and use more one point if we need that.

Now we will derive finite-difference formula at *i=1*, using algorithm (1.2) by applying boundary condition to get

$$y_1' = \frac{-2y_1+2y_2+hy_0'}{3h} - \frac{5h^2}{18}y^{(''')}(\xi_1) \qquad \dots(2.65)$$

$$y_1'' = \frac{-4y_1+2y_2+2y_3-6hy_0'}{11h^2} - \frac{29h^2}{132}y^{(iv)}(\eta_1) \qquad \dots(2.66)$$

$$y_1''' = \frac{53y_1-96y_2+51y_3-8y_4+18hy_0'}{25h^2} + \frac{53h^2}{500}y^{(v)}(\zeta_1) \qquad \dots(2.67)$$

By substituting equations (2.65), (2.66) and (2.67) in Eq.(2.59) we get

$$\frac{53y_1-96y_2+51y_3-8y_4+18hy_0'}{25h^2} = p_2(x_1)\left[\frac{-4y_1+2y_2+2y_3-6hy_0'}{11h^2}\right] + p_1(x_1)\left[\frac{-2y_1+2y_2+hy_0'}{3h}\right] + p_0(x_1)y_1 + r(x_1) - \frac{h^2}{49500}[13750p_1(x_1)y'''(\xi_1) + 10875p_2(x_1)y^{(iv)}(\eta_1) + 5247y^{(v)}(\zeta_1)] \qquad \dots(2.68)$$

By omitting the error term and multiplying the previous equation by $825h^3$ we get the finite-difference formula at *i*=1

$$[1749 + 300hp_2(x_1) + 550h^2p_1(x_1) - 825h^3p_0(x_1)]w_1 - [3168 + 150hp_2(x_1) + 550h^2p_1(x_1)]w_2 + [1682 - 150hp_2(x_1)]w_3 - 264w_4 + [594 + 450hp_2(x_1) - 275h^2p_1(x_1)]hw_0' = 825h^3r(x_1) \qquad \dots(2.69)$$

Now we will derive the formula at *i=N-1*using boundary conditions

$$y_{N-1}''' = \frac{-5y_{N-2}+18y_{N-1}-27y_N+14y_{N+1}-6hy_{N+1}'}{3h^3} + \frac{3h^2}{20}y^{(v)}(\zeta_{N-1}) \qquad \dots(2.70)$$

Then substituting equations (2.26),(2.27) the centered finite-difference formula for y', y'' and (2.70) in (2.69) we get

$$\frac{-5y_{N-2}+18y_{N-1}-27y_N+14y_{N+1}-6hy'_{N+1}}{3h^3} = p_2(x_{N-1})\left[\frac{y_{N-2}-2y_{N-1}+y_N}{h^2}\right] + p_1(x_{N-1})\left[\frac{y_N-y_{N-2}}{2h}\right] + p_0(x_{N-1})y_{N-1} + r(x_{N-1}) - \frac{h^2}{60}[20p_1(x_{N-1})y'''(\xi_{N-1}) + 5p_2(x_{N-1})y^{(iv)}(\eta_{N-1}) + 3y^{(v)}(\zeta_{N-1})] \quad ...(2.71)$$

By omitting the error term and multiplying the previous equation by $6h^3$ we get the finite-difference formula at *i=N-1*

$$-[10 + 6hp_2(x_{N-1}) - 3h^2p_1(x_{N-1})]w_{N-2} + [36 + 12hp_2(x_{N-1}) - 6h^3p_0(x_{N-1})]w_{N-1} - [54 + 6hp_2(x_{N-1}) + 3h^2p_1(x_{N-1})]w_N + 28w_{N+1} - 12hw'_{N+1} = 6h^3r(x_{N-1}) \quad ...(2.72)$$

In order to maintain second order convergence, we derive a new finite-difference approximation for third derivative using the first derivative at *i=N* and one more point using algorithm (1.2) as following:

$$y'''_N = \frac{-y_{N-2}+9y_N-8y_{N+1}+6hy'_{N+1}}{3h^3} - \frac{3h^2}{20}y^{(v)}(\zeta_N) \quad ...(2.73)$$

By substituting equations (2.31), (2.32) and (2.73) in Eq.(2.59) we get

$$\frac{-y_{N-2}+9y_N-8y_{N+1}+6hy'_{N+1}}{3h^3} = p_2(x_N)\left[\frac{y_{N-1}-2y_N+y_{N+1}}{h^2}\right] + p_1(x_N)\left[\frac{y_{N+1}-y_{N-1}}{2h}\right] + p_0(x_N)y_N + r(x_N) - \frac{h^2}{60}[10p_1(x_1)y'''(\xi_1) + 5p_2(x_1)y^{(iv)}(\eta_1) - 3y^{(v)}(\zeta_1)] \quad ...(2.74)$$

By omitting the error term and multiplying the previous equation by $6h^3$ we get the finite-difference formula at *i=N*

$$-2w_{N-2} - [6hp_2(x_N) - 3h^2p_1(x_N)]w_{N-1} + [18 + 12hp_2(x_N) - 6h^3p_0(x_N)]w_N - [16 + 6hp_2(x_N) + 3h^2p_1(x_N)]w_{N+1} + 12hw'_{N+1} = 6h^3r(x_N) \quad ...(2.75)$$

Using equations (2.64), (2.69), (2.72) and (2.75) together gives a linear system with $N\times N$ nearly penta-diagonal matrix problem

$$Aw = \mathrm{c} \qquad \ldots(2.76)$$

Where $A=$

$$\begin{bmatrix} 1749-G_1 & -(3168+G_2) & 1683-G_3 & -264 & & 0 \\ -(3+F_1(x_2)) & 10-F_2(x_2) & -(12+F_3(x_2)) & 6 & -1 & \\ & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & & -(3+F_1(x_{N-2})) & 10-F_2(x_{N-2}) & -(12+3F_3(x_{N-2})) \\ & & & -(10+3F_1(x_{N-1})) & 36-3F_2(x_{N-1}) & -(54+3F_3(x_{N-1})) \\ 0 & & & -2 & -3F_1(x_N) & (18-3F_2(x_N)) \end{bmatrix}_{N\times N}$$

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{N-1} \\ w_N \end{bmatrix}_{N\times 1} \quad ; \quad c = \begin{bmatrix} (3+G_0)h\alpha_1 + 825h^3 r(x_1) \\ 2h^3 r(x_2) \\ \vdots \\ \beta_0 + 2h^3 r(x_{N-2}) \\ -28\beta_0 + 12h\beta_1 + 6h^3 r(x_{N-1}) \\ (16+3F_3(x_N))\beta_0 - 12h\beta_1 + 6h^3 r(x_N) \end{bmatrix}_{N\times 1}$$

$$F_1(x_i) = 2hp_2(x_i) - h^2 p_1(x_i)\ ,\ F_2(x_i) = -4hp_2(x_i) + 2h^3 p_0(x_i)$$

$$F_3(x_i) = 2hp_2(x_i) + h^2 p_1(x_i)\ ,\text{for } i=2,3,\ldots,N$$

$$G_0 = -450hp_2(x_1) + 275h^2 p_1(x_1)$$

$$G_1 = -300hp_2(x_1) - 550h^2 p_1(x_1) + 825h^3 p_0(x_1)$$

$$G_2 = 150hp_2(x_1) + 550h^2 p_1(x_1)\ ,G_3 = -450hp_2(x_1)$$

Now, using LU-decomposition with MATLAB, we will solve the linear system (2.36), to reduce calculations at several values of step-size (h=0.1, 0.05, 0.025), then we will extrapolate the solution to get higher-order accuracy.

Algorithm (2.4): Linear Finite-Difference Method (2.4) for Solving Third-Order BVPS Case (IV): See full MATLAB program (p.208-212)

To approximate the solution for the boundary-value problem

$$y^{(3)} = p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \qquad a \le x \le b$$

with boundary conditions $y'(a) = \alpha_1,\ y(b) = \beta_0,\ y'(b) = \beta_1$

This algorithm has the same steps in algorithm (2.1) except modification in steps (5) &(6) as:

Step(5): To determine A matrix diagonals in the system (2.79); For i=1,2,3,…,N_k set

(The main diagonal for A (d0))

d0=[$(1749 - G_1)$ 10-$F_2(2)$ …..(36-3$F_2(N_k-1)$) (18-3$F_2(N_k)$)];

(The first upper diagonal for A (dU1))

dU1=[$-(3168 + G_2)$ -(12+$F_3(2)$) ……. -(54+3$F_3(N_k-1)$)];

(The second upper diagonal for A (dU2))

dU2=$[1683 - G_3\ 6 \ldots 6]_{1\times(N_k-2)}$

(The third upper diagonal for A (dU3)),dU3=$[-264 \quad -1 \ \ldots\ -1]_{1\times(N_k-3)}$

(The first lower diagonal for A (dL1))

dL1=[-(3+$F_1(2)$) -(3+$F_1(3)$) ……(36-$F_1(N_k-1)$ -$F_1(N_k)$] ;

(The second lower diagonal for A (dL2))

dL2= $[\,0\ \ldots\ 0\ \ -2]_{1\times(N_k-2)}$

(The constant vector on (c) Eq.(2.76))

c=$[((3+G_0)h\alpha_1+825h^3r(x_1))\ ;\ 2h^3r(3);\ldots;(2h^3r(N_k-2)+b_0\ )\ ;(\ -28\beta_0+12h\beta_1+6h^3r(x_{N_k-1}))\ ;(\ (16+3F_3(x_{N_k}))\beta_0-12h\beta_1+6h^3r(x_{N_k}))]$

Step (6): Factorize A into LU, where L is lower tri-diagonal matrix with unit main diagonal and U is upper four-diagonal.

Then we complete the other steps in algorithm to get the results.

Example 4: Consider the linear third-order boundary value problem

$$y'''(x)=2x^2y''-3xy'-5x^2y+e^{2x}(3x^3-x^2-5x-4), 0\le x\le 1$$

with boundary conditions : $y'(0)=1,\quad y(1)=0,\quad y'(1)=-e^2$

So $p_2(x)=2x^2$, $p_1(x)=-3x$, $p_0(x)=-5x^2$, and

$r(x)=e^{2x}(3x^3-x^2-5x-4)$ on the interval [*0,1*]

Now by applying algorithm (2.4) to the example we get the following results that approximated solutions at several values of h as listed in Table (2.8)

Table 2.8

x_i	Exact Y	w_i(h=0.1)	w_i(h=0.05)	w_i(h=0.025)	w_i(h=0.0125)	Ext3(0.1)	Ext4(0.1)
0.1	1.09926248	1.05898030	1.08713743	1.09617385	1.09849834	1.09936351	1.09927765
0.2	1.19345976	1.15575555	1.18203872	1.19054483	1.19273820	1.19355222	1.19347404
0.3	1.27548316	1.24190126	1.26518109	1.27284419	1.27482924	1.27556239	1.27549605
0.4	1.33532456	1.30710705	1.32649177	1.33304879	1.33475972	1.33538653	1.33533560
0.5	1.35914091	1.33713767	1.35204337	1.35729604	1.35868187	1.35918259	1.35914976
0.6	1.32804677	1.31259711	1.32283842	1.32667453	1.32770399	1.32806663	1.32805320
0.7	1.21655999	1.20734691	1.21324331	1.21566698	1.21633546	1.21655928	1.21656392
0.8	0.99060648	0.98651200	0.98897681	0.99015092	0.99049051	0.99059188	0.99060806
0.9	0.60496475	0.60404784	0.60453932	0.60483719	0.60493129	0.60495203	0.60496460

The previous results reveal that as h decreases the accuracy increases and the maximum error for each approximation decreases as listed in Table (2.9).

Table 2.9

$\lvert Y - W \rvert$	**Maximum Error**
$\lvert Y - w_i(h = 0.1) \rvert$	4.03×10^{-2}
$\lvert Y - w_i(h = 0.05) \rvert$	1.21×10^{-2}
$\lvert Y - w_i(h = 0.025) \rvert$	3.1×10^{-3}
$\lvert Y - w_i(h = 0.0125) \rvert$	7.4614×10^{-4}
$\lvert Y - Ext3_i(h = 0.1) \rvert$	1.0696×10^{-5}
$\lvert Y - Ext4_i(h = 0.1) \rvert$	1.5167×10^{-5}

This table reveals that the results of this method are better than method (2.3), but method (2.1) and (2.2) are better than method (2.4).

Table 2.10

Max Error for	**Boundary Conditions for Third-Order Boundary-Value Problems** Case (I)	Case (II)	Case (III)	Case (IV)
	$y(a) = \alpha_0$ $y'(a) = \alpha_1$ $y(b) = \beta_0$	$y(a) = \alpha_0$ $y(b) = \beta_0$ $y''(b) = \beta_2$	$y(a) = \alpha_0$ $y'(a) = \alpha_1$ $y''(b) = \beta_2$	$y'(a) = \alpha_1$ $y(b) = \beta_0$ $y'(b) = \beta_1$
$\lvert Y - w_i(h = 0.1) \rvert$	7.7×10^{-3}	7.9792×10^{-3}	3.1646	4.03×10^{-2}
$\lvert Y - w_i(h = 0.05) \rvert$	2.4×10^{-3}	5.9320×10^{-3}	2.4004	1.21×10^{-2}
$\lvert Y - w_i(h = 0.025) \rvert$	6.4415×10^{-4}	2.0350×10^{-3}	1.2252	3.1×10^{-3}
$\lvert Y - w_i(h = 0.0125) \rvert$	1.6704×10^{-4}	5.8377×10^{-4}	0.4382	7.4614×10^{-4}
$\lvert Y - Ext3_i(h = 0.1) \rvert$	3.2016×10^{-5}	1.0003×10^{-4}	7.460×10^{-1}	1.0696×10^{-5}
$\lvert Y - Ext4_i(h = 0.1) \rvert$	1.6704×10^{-6}	4.7850×10^{-5}	1.224×10^{-1}	1.5167×10^{-5}

Now we will discuss the boundary conditions at each case and their effects in method accuracy.

Table (2.10) shows that the bad approximations are in the two cases (III) & (IV), where β_0 is absence in case (III) and α_0 is absence in case (IV). And case (I) approximation is better than case (II) approximation, where the order of derivative in boundary condition in case (I) is one and is two in case (II). So we can conclude that the order of derivatives in boundary conditions and the absence of initial value for problem at endpoints interval affect and control the method accuracy.

So this explains when we can say that the method has good results or not.

CHAPTER THREE

Fourth-Order Linear Boundary-Value Problem

This chapter is concerned with the finite-difference methods for solving linear fourth-order boundary-value problems, that have many applications in physics and engineering especially the famous case for the equilibrium states of a beam-column and others several cases of beam vibrations.

There are many cases for the linear fourth-order boundary-value problem that differ as the boundary conditions differ, which their solution existence and uniqueness are discussed widely by several papers (see [6]).

In this thesis we will discuss the solution approximations for three cases of fourth-order boundary-value problem using some finite-difference method.

3.1 Finite-Difference Method (3.1) for Solving Fourth-Order Boundary Value Problem:

Consider the linear fourth-order boundary-value problem:

$$y^{(iv)} = p_3(x)y''' + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \quad a \leq x \leq b \qquad \ldots(3.1)$$

with first especial case of four boundary conditions:

$$y(a) = \alpha_0 \,, y''(a) = \alpha_2 \,, y(b) = \beta_0 \,, y'(b) = \beta_1 \qquad \ldots(3.2)$$

First, we select an integer $N>0$ and divide the interval $[a,b]$ into $(N+1)$ equal subintervals whose endpoints are the mesh points $x_i = a + ih$ for $i=0,1,2,\ldots,N+1$,where the step size $h = (b-a)/(N+1)$.

Second, we approximate all derivatives in Eq.(3.1) using finite-difference approximation on the interior mesh points.

$$y'(x_i) = \frac{y(x_{i+1}) - y(x_{i-1})}{2h} - \frac{h^2}{6} y'''(\xi_i) \qquad \ldots(3.3)$$

$$y''(x_i) = \frac{y(x_{i-1}) - 2y(x_i) + y(x_{i+1})}{h^2} - \frac{h^2}{12} y^{(iv)}(\eta_i) \qquad \ldots(3.4)$$

$$y'''(x_i) = \frac{-y(x_{i-2}) + 2y(x_{i-1}) - 2y(x_{i+1}) + y(x_{i+2})}{2h^3} - \frac{h^2}{4} y^{(v)}(\zeta_i) \qquad \ldots(3.5)$$

$$y^{(iv)}(x_i) = \frac{y(x_{i-2}) - 4y(x_{i-1}) + 6y(x_i) - 4y(x_{i+1}) + y(x_{i+2})}{h^4} - \frac{h^2}{6} y^{(vi)}(\varphi_i) \qquad \ldots(3.6)$$

Those are centered-difference formulas with second order convergence.

Now substituting the equations (3.3), (3.4), (3.5) and (3.6) in Eq. (3.1) we get the equation

$\frac{y(x_{i-2})-4y(x_{i-1})+6y(x_i)-4y(x_{i+1})+y(x_{i+2})}{h^4} =$
$p_3(x_i)[\frac{-y(x_{i-2})+2y(x_{i-1})-2y(x_{i+1})+y(x_{i+2})}{2h^3}] + p_2(x_i)[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}] +$
$p_1(x_i)[\frac{y(x_{i+1})-y(x_{i-1})}{2h}] + p_0(x_i)y(x_i) + r(x_i) - \frac{h^2}{12}[2p_1(x_i)y'''(\xi_i) +$
$p_2(x_i)y^{(iv)}(\eta_i) + 3p_3(x_i)y^{(v)}(\eta_i) + 2y^{(vi)}(\varphi_i)]$ … (3.7)

A Finite-Difference method with second order truncation error results by using this equation together with the boundary conditions as

$$w_0 = \alpha_0 \,, w_0'' = \alpha_2 \,, w_{N+1} = \beta_0 \,, w_0' = \beta_1$$

$\frac{w_{i-2}-4w_{i-1}+6w_i-4w_{i+1}+y(x_{i+2})}{h^4} - p_3(x_i)\left[\frac{-w_{i-2}+2w_{i-1}-2w_{i+1}+w_{i+2}}{2h^3}\right] -$
$p_2(x_i)\left[\frac{w_{i-1}-2w_i+w_{i+1}}{h^2}\right] - p_1(x_i)\left[\frac{w_{i+1}-w_{i-1}}{2h}\right] - p_0(x_i)w_i = r(x_i)$ … (3.8)

for *i=2,3,4,…,N-1*

Now by multiplying Eq.(3.8) by $2h^4$ it can be written as :

$[2 + hp_3(x_i)]w_{i-2} - [8 + 2hp_3(x_i) + 2h^2p_2(x_i) - h^3p_1(x_i)]w_{i-1} +$
$[12 + 4h^2p_2(x_i) + h^4p_0(x_i)]w_i - [8 - 2hp_3(x_i) + 2h^2p_2(x_i) +$
$h^3p_1(x_i)]w_{i+1} + [2 - hp_3(x_i)]w_{i+2} = 2h^4r(x_i)$ …(3.9)

But this formula is suitable for N-2 mesh points, so we need to derive special formulas at *i*=1, *N* using the boundary conditions to define finite-difference approximation for third and fourth derivatives

$$y_1' = \frac{y_2-y_0}{2h} - \frac{h^2}{6}y'''(\xi_1) \quad \text{…(3.10)}$$

$$y_1'' = \frac{y_0-2y_1+y_2}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_1) \quad \text{…(3.11)}$$

$$y_1''' = \frac{y_0+3y_1-9y_2+5y_3-6h^2y_0''}{11h^3} - \frac{9h^2}{44}y^{(v)}(\zeta_1) \quad \text{…(3.12)}$$

In order to maintain second order convergence we derive a new finite-difference approximation for fourth order derivative using the boundary conditions at i=0 and one more point using algorithm(1.2), which is given by

$$y_1^{(iv)} = \frac{-25y_0+64y_1-54y_2+16y_3-y_4+12h^2y_0''}{10h^4} - \frac{2h^2}{25}y^{(vi)}(\varphi_1) \qquad \ldots(3.13)$$

By substituting equations (3.10),(3.11),(3.12) and (3.13) on Eq.(3.1) we get

$$\frac{-25y_0+64y_1-54y_2+16y_3-y_4+12h^2y_0''}{10h^4} = p_3(x_1)[\frac{y_0+3y_1-9y_2+5y_3-6h^2y_0''}{11h^3}] + p_2(x_1)\left[\frac{y_0-2y_1+y_2}{h^2}\right] + p_1(x_1)\left[\frac{y_2-y_0}{2h}\right] + p_0(x_1)y_1 + r(x_1) - \frac{h^2}{3300}[550p_1(x_1)y'''(\xi_1) + 275p_2(x_1)y^{(iv)}(\eta_1) + 675y^{(v)}(\zeta_1) - 264y^{(vi)}(\varphi_1)] \qquad \ldots(3.14)$$

By omitting the error term and multiplying the previous equation by $110h^4$ we get the finite-difference formula at i=1

$$-[275 + 10hp_3(x_1) + 110h^2p_2(x_1) - 55h^3p_1(x_1)]w_0 + [704 - 30hp_3(x_1) + 220h^2p_2(x_1) - 110h^4p_0(x_1)]w_1 - [594 - 90hp_3(x_1) + 110h^2p_2(x_1) + 55h^3p_1(x_1)]w_2 + [176 - 50hp_3(x_1)]w_3 - 11w_4 + [132 + 60hp_3(x_1)]h^2w_0'' = 110h^4r(x_1) \qquad \ldots(3.15)$$

To find the suitable finite-difference formula at i=N we follow the same step as at i=1 as

$$y_N' = \frac{y_{N+1}-y_{N-1}}{2h} - \frac{h^2}{6}y'''(\xi_N) \qquad \ldots(3.16)$$

$$y_N'' = \frac{y_{N-1}-2y_N+y_{N+1}}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_N) \qquad \ldots(3.17)$$

$$y_N''' = \frac{-y_{N-2}+9y_{N-1}-8y_{N+1}-6hy_{N+1}'}{3h^3} - \frac{3h^2}{20}y^{(v)}(\zeta_N) \qquad \ldots(3.18)$$

$$y_N^{(iv)} = \frac{-3y_{N-3}+32y_{N-2}-108y_{N-1}+192y_N-113y_{N+1}+60hy'_{N+1}}{12h^4} - \frac{h^3}{21}y^{(vii)}(\varphi_N) \quad \ldots(3.19)$$

By substituting equations (3.16), (3.17), (3.18) and (2.19) in Eq.(3.1) we get

$$\frac{-3y_{N-3}+32y_{N-2}-108y_{N-1}+192y_3-113y_4+60hy'_{N+1}}{12h^4} = p_3(x_N)\left[\frac{-y_{N-2}+9y_{N-1}-8y_{N+1}-6hy'_{N+1}}{3h^3}\right] +$$
$$p_2(x_N)\left[\frac{y_{N-1}-2y_N+y_{N+1}}{h^2}\right] + p_1(x_N)\left[\frac{y_{N+1}-y_N}{2h}\right] + p_0(x_N)y_N + r(x_N) -$$
$$\frac{h^2}{420}[70p_1(x_N)y'''(\xi_N) + 35p_2(x_2)y^{(iv)}(\eta_N) + 63y^{(v)}(\zeta_N) - 20hy^{(vii)}(\varphi_N)] \quad \ldots(3.20)$$

By omitting the error term and multiplying the previous equation by $12h^4$ we get the finite-difference formula at *i*=N

$$-3w_{N-3} + [32 + 4hp_3(x_N)]w_{N-2} - [108 - 12h^2p_2(x_N) - 6h^3p_1(x_N)]w_{N-1} +$$
$$[192 - 36hp_3(x_N) + 24h^2p_2(x_N) - 12p_0(x_N)]w_N - [113 - 32hp_3(x_N) +$$
$$12h^2p_2(x_N) + 6h^3p_1(x_N)]w_{N+1} + [60 - 24hp_3(x_N)]hw'_{N+1} = 12h^4r(x_N)$$

…(3.21)

Using equations (3.9), (3.15), and (3.21) together give a system with *N*×*N* seven-diagonal matrix problem

$$Aw = \text{c} \quad \ldots(3.22)$$

where

A=

$$\begin{bmatrix} 704 - G_1 & -(594 + G_2) & 176 - G_3 & -11 & & & & \\ -(8 + F_2(x_2)) & 12 - F_3(x_2) & -(8 + F_4(x_2)) & 2 + F_1(x_2) & & & & \\ 2 - F_1(x_3) & -(8 + F_2(x_3)) & 12 - F_3(x_3) & -(8 + F_4(x_3)) & 2 + F_1(x_3) & & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & & & & 2 - F_1(x_{N-1}) & -(8 + F_2(x_{N-1})) & 12 - F_3(x_{N-1}) & -(8 + F_4(x_{N-1})) \\ & & & & -3 & 32 - G_{N2} & -(108 + G_{N1}) & 192 - G_N \end{bmatrix}_{N\times N}$$

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \\ \vdots \\ \\ w_{N-1} \\ w_N \end{bmatrix}_{N\times 1}, c = \begin{bmatrix} (275+G_0)\alpha_0 - (132+60hp_3(x_1))h^2 + 110h^4 r(x_1) \\ (-2+F_1(x_2))\alpha_0 + 2h^4 r(x_2) \\ 2h^4 r(x_3) \\ \vdots \\ 2h^4 r(x_{N-2}) \\ -(2+F_1(x_{N-1}))\beta_0 + 2h^4 r(x_{N-1}) \\ (113+G_{N0})\beta_0 - \big(60-24hp_3(x_N)\big)h\beta_1 + 12h^4 r(x_N) \end{bmatrix}_{N\times 1}$$

$F_1(x_i) = -hp_3(x_i) \quad , F_2(x_i) = 2hp_3(x_i) + 2h^2 p_2(x_i) - h^3 p_1(x_i)$

$F_3(x_i) = -4h^2 p_2(x_i) + 2h^4 p_0(x_i) \ , F_4(x_i) = -2hp_3(x_i) + 2h^2 p_2(x_i) + h^3 p_1(x_i)$

$G_0 = 10hp_3(x_1) + 110h^2 p_2(x_1) - 55h^3 p_1(x_1),$

$G_1 = 30hp_3(x_1) - 220h^2 p_2(x_1) + 110h^4 p_0(x_1)$

$G_2 = -90hp_3(x_1) + 110h^2 p_2(x_1) + 55h^3 p_1(x_1), G_3 = 50hp_3(x_1)$

$G_{N2} = -4hp_3(x_N) \quad , G_{N1} = 12h^2 p_2(x_N) - 6h^3 p_1(x_N)$

$G_N = 36hp_3(x_N) - 24h^2 p_2(x_N) + 12h^4 p_0(x_N)$

$G_{N0} = -32hp_3(x_N) + 12h^2 p_2(x_N) + 6h^3 p_1(x_N)$

Now, we use the LU-decomposition method for solving the linear system (3.22) at several values of step-size (h), and then we use Extrapolation method to get more accuracy for the method without more calculations.

Algorithm (3.1): Linear Finite-Difference Method (3.1) for Solving Fourth-Order BVPS Case (I): See full MATLAB program (p.213-218)

To approximate the solution of the boundary value problem

$y^{(iv)} = p_3(x)y''' + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \quad a \le x \le b$

with four boundary conditions: $y(a) = \alpha_0 \, , y''(a) = \alpha_2 \, , y(b) = \beta_0 \, , y'(b) = \beta_1$

Step(1):Input endpoints a,b; boundary conditions $\alpha_0, \beta_0, \alpha_2, \beta_1$

Step(2): For *k*=1,2,3,4 (To determine h=0.1,0.05,0.025,0,0125)

- set $N_k=(10*2^{(k-1)})-1$;
- $h=(b-a)/(N_k+1)$;
- do steps(2-9)

Step(3): For i=1,2,3, …,$N(k)$ set $x_i=a+ih$

find $p_0(x_i), p_1(x_i), p_2(x_i), p_3(x_i) and\ r(x_i)$.

(The values of associated functions on Eq.(2.1)

Step(4): For $i=1,2,3,\ldots,N_k$ set

$F_1(i) = -hp_3(i) \quad , F_2(i) = 2hp_3(i) + 2h^2p_2(i) - h^3p_1(i)$

$F_3(i) = -4h^2p_2(i) + 2h^4p_0(i) \ , F_4(i) = -2hp_3(i) + 2h^2p_2(i) + h^3p_1(i)$

(The adding Eq.(3.3),(3.4)&(3.5)finite-difference formula for y''' ,y'', y' as F's)

$G_0 = 10hp_3(1) + 110h^2p_2(1) - 55h^3p_1(1) \ , G_1 = 30hp_3(1) - 220h^2p_2(1) + 110h^4p_0(1)$

$G_2 = -90hp_3(1) + 110h^2p_2(1) + 55h^3p_1(1) \ , G_3 = 50hp_3(1)$

(The adding Eq.(3.10),(3.11)&(3.12)finite-difference formula for y''' ,y'', y' as Gi's)

$G_{N2} = -4hp_3(N_k) \quad , G_{N1} = 12h^2p_2(N_k) - 6h^3p_1(N_k)$

$G_N = 36hp_3(N_k) - 24h^2p_2(N_k) + 12h^4p_0(N_k)$

$G_{N0} = -32hp_3(N_k) + 12h^2p_2(N_k) + 6h^3p_1(N_k)$

(The adding Eq.(3.16),(3.17) &(3.18) finite-difference formula for y''' ,y'' and y' as G_{Ni}'s)

Step(5): To determine the diagonals for matrix *A* in the system (3.22) ; For *i*=1,2,3,…,*N(k)* set (The main diagonal for A (d0))

d0=[(704-G_1) (12-$F_3(2)$) ….. (12 – $F_3(N_k$-1)) (192+G_N)];

(The first upper diagonal for A (dU1))

dU1=[-(594+G_2) -(8+$F_4(2)$) ……. -(8-$F_4(N_k$-1))];

(The second diagonal for A (dU2))

dU2=[(176-G_3) (2+$F_1(2)$) …. (2+$F_2(N_k$-2))]

(The third diagonal for A (dU3)) dU3=$[-11\ 0\ \dots\ 0]_{1\times(N_k-3)}$

(The first lower diagonal for A (dL1))

dL1=$[-(8+F2(2))\ -(8+F2(3))\ \dots\dots -(8+F2(N_k-1)$ – $(108+G_{N1})]_{1\times(N_k-1)}$]

(The second lower diagonal for A (dL2))

dL2= $[(2-F1(3))\ \dots\ 2-F1(N_k-1)\ (32-G_{N2})]_{1\times(N_k-2)}$

(The third lower diagonal for A (dL3)) dL3=$[0\ \dots\ 0\ -3]_{1\times(N_k-3)}$

(The constant vector on (c) Eq.(3.22))

c=[$(110h^4r(1) + (275 - G_0)a0 - (132 + 60hp_3(1))h^2$a2) ;$(2h^4r(2) + (-2 + F_1(2))a0)$; $2h^4r(3)$; …..;$2h^4r(N_k - 1) - (2 + F_1(N_k - 1))$; $2h^4r(N_k) - (113 - G_{N0})\beta_0 - (60 - 24hp_3(N_k)hb1]$

Step(6): Factorize A into LU, where L is lower four-diagonal matrix with unit main diagonal and U is upper tridiagonal.

Step(7) For $i=1,2,3,\ldots,N_k$

- Solve LY=d to find Y using forward substitution

Step(8) For $i=N_k-1 ,.. 2,1$

- Solve U*w*=Y to find the approximation solution w using backward substitution

Step(9) Set N_1=9 For $i=1,2,3,\ldots N_k$, for j=1, 2, 4, …,N_k set $W1_{i,k}=w_j$;

Step(10) For $i=1,2,\ldots,N1$ set h=(b-a)/N_1 ; x(i)=a+ih

Step(11) Extrapolate the solution at h= .1,0.05,0.025,0,0125

Ext2(h=0.1)=(4*W1(h=0.05)-W1(h=0.1))/3;

Ext2(h=0.05)=(4*W1(h=0.025)-W1(h=0.05))/3;

Ext2(h=0.025)=(4*W1(h=0.0125)-W1(h=0.025))/3;

Ext3(h=0.1)=(16*Ext2(h=0.05)-Ext1(h=0.1))/15;

Ext3(h=0.05)=(16*Ext2(h=0.025)-Ext2(h=0.05))/15;

Ext4=(64*Ext3(h=0.05)-Ext3(h=0.1))/63;

Step(12) Output [x' ,W1',Ext4] (The approximation solutions and Extrapolation)

Example 1: Consider the linear fourth-order boundary value problem

$y^{(iv)}(x) = (1 + 2x^2)y''' - 3xy'' - 5x^2y' + (x^3 + 1)y + e^{2x}(x^4 + 5x^3 + x^2 - 7x - 13),$

on $0 \leq x \leq 1$ with boundary conditions:

$$y(0) = 1, \quad y(1) = 0, \quad y'(0) = 1\,, \quad y''(1) = -4e^2$$

So $p_3(x) = (1 + 2x^2)$, $p_2(x) = -3x$, $p_1(x) = -5x^2$, $p_0(x) = x^3 + 1$, and $r(x) = e^{2x}(x^4 + 5x^3 + x^2 - 7x - 13)$ on the interval $[0,1]$

Now by applying algorithm (3.1) to the example we get the following results that approximate the solutions at several values of h as listed in Table (3.1)

Table 3.1

X_i	Exact(Y)	w(h=0.1)	w(h=0.05)	w(h=0.025)	w(h=0.0125)	Ext3(h=0.1)	Ext4(h=0.1)
0.1	**1.09926248**	**1.09961120**	**1.09935462**	**1.09928575**	**1.09926831**	**1.09926237**	**1.09926248**
0.2	**1.19345976**	**1.19411723**	**1.19363232**	**1.19350327**	**1.19347065**	**1.19345955**	**1.19345976**
0.3	**1.27548316**	**1.27636680**	**1.27571399**	**1.27554130**	**1.27549772**	**1.27548289**	**1.27548316**
0.4	**1.33532456**	**1.33632049**	**1.33558390**	**1.33538983**	**1.33534089**	**1.33532425**	**1.33532455**
0.5	**1.35914091**	**1.36011973**	**1.35939540**	**1.35920492**	**1.35915693**	**1.35914060**	**1.35914091**
0.6	**1.32804677**	**1.32888386**	**1.32826449**	**1.32810152**	**1.32806047**	**1.32804647**	**1.32804676**
0.7	**1.21655999**	**1.21716008**	**1.21671664**	**1.21659938**	**1.21656985**	**1.21655973**	**1.21655999**
0.8	**0.99060648**	**0.99093068**	**0.99069209**	**0.99062803**	**0.99061188**	**0.99060629**	**0.99060648**
0.9	**0.60496475**	**0.60505621**	**0.60498997**	**0.60497113**	**0.60496634**	**0.60496464**	**0.60496474**

The previous results reveal that as h decreases the accuracy increases and that the maximum errors in approximations are as listed in Table (3.2)

Table 3.2

$\lvert Y-W\rvert$	**Maximum Error**
$\lvert Y-w_i(h=0.1)\rvert$	9.9593×10^{-4}
$\lvert Y-w_i(h=0.05)\rvert$	2.5934×10^{-4}
$\lvert Y-w_i(h=0.025)\rvert$	6.5268×10^{-5}
$\lvert Y-w_i(h=0.0125)\rvert$	1.6338×10^{-5}
$\lvert Y-Ext3_i(h=0.1)\rvert$	3.1646×10^{-7}
$\lvert Y-Ext4_i(h=0.1)\rvert$	4.5419×10^{-9}

From previous table we can say that this method has good approximation results for the problem.

3.2 Finite-Difference Method (3.2) for Solving Fourth-Order Boundary Value Problem:

Consider the linear fourth-order boundary-value problem:

$$y^{(iv)} = p_3(x)y''' + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \quad a \le x \le b \qquad \ldots(3.23)$$

with second especial case of four boundary conditions:

$$y(a) = \alpha_0\,, y'(a) = \alpha_1\,, y(b) = \beta_0\,, y'(b) = \beta_1 \qquad \ldots(3.24)$$

As in previous section ,we select an integer *N*>0 and divide the interval [*a,b*] into (*N*+1) equal subintervals whose endpoints are the mesh points $x_i = a + ih$ for *i*=0,1,2,…,*N+1*,where the step size $h = (b-a)/(N+1)$.

Second, because of symmetry of the fourth-order derivate we will use the finite difference formula Eq.(3.9) in pervious section for N=2,3,…N-1,

$$[2 + hp_3(x_i)]w_{i-2} - [8 + 2hp_3(x_i) + 2h^2p_2(x_i) - h^3p_1(x_i)]w_{i-1} + [12 + 4h^2p_2(x_i) + h^4p_0(x_i)]w_i - [8 - 2hp_3(x_i) + 2h^2p_2(x_i) + h^3p_1(x_i)]w_{i+1} + [2 - hp_3(x_i)]w_{i+2} = 2h^4r(x_i) \qquad \ldots(3.9)$$

that is centered-difference formula with second order convergence.

But this formula is suitable for N-2 mesh points, so we need to derive special formula when i=1,N using the boundary conditions to define finite-difference approximation for third and fourth derivatives

$$y_1''' = \frac{8y_0-9y_1+y_3+6hy_0'}{3h^3} - \frac{3h^2}{20}y^{(v)}(\zeta_1) \qquad \ldots(3.25)$$

In order to maintain second order convergence, we derive a new finite-difference approximation for fourth order derivative using the boundary conditions at i=0 and one more point using (1.2) which is given by

$$y_1^{(iv)} = \frac{-113y_0+192y_1-108y_2+32y_3-3y_4-60hy_0'}{12h^4} + \frac{h^3}{21}y^{(vii)}(\varphi_1) \quad \ldots(3.26)$$

By substituting equations (3.10), (3.11), (3.25) and (3.26) in Eq.(3.23) we get the equation

$$\frac{-113y_0+192y_1-108y_2+32y_3-3y_4-60hy_0'}{12h^4} = p_3(x_1)[\frac{8y_0-9y_1+y_3+6hy_0'}{3h^3}] +$$
$$p_2(x_1)\left[\frac{y_0-2y_1+y_2}{h^2}\right] + p_1(x_1)\left[\frac{y_2-y_0}{2h}\right] + p_0(x_1)y_1 + r(x_1) - \frac{h^2}{420}[70p_1(x_1)y'''(\xi_1) +$$
$$35p_2(x_1)y^{(iv)}(\eta_1) + 147y^{(v)}(\zeta_1) + 20hy^{(vii)}(\varphi_1)] \qquad \ldots(3.27)$$

By omitting the error term and multiplying the previous equation by $12h^4$ we get the finite-difference formula at i=1

$$-[113 + 32hp_3(x_1) + 12h^2p_2(x_1) - 6h^3p_1(x_1)]w_0 + [192 + 36hp_3(x_1) +$$
$$24h^2p_2(x_1) - 12h^4p_0(x_1)]w_1 - [108 + 12h^2p_2(x_1) + 6h^3p_1(x_1)]w_2 +$$
$$[32 - 4hp_3(x_1)]w_3 - 3w_4 - [60 + 24hp_3(x_1)]hw_0' = 12h^4r(x_1) \qquad \ldots(3.28)$$

and by using finite-difference formula (3.21) at i=N in the previous section

$-3w_{N-3} + [32 + 4hp_3(x_N)]w_{N-2} - [108 - 12h^2p_2(x_N) - 6h^3p_1(x_N)]w_{N-1} +$
$[192 - 36hp_3(x_N) + 24h^2p_2(x_N) - 12p_0(x_N)]w_N - [113 - 32hp_3(x_N) +$
$12h^2p_2(x_N) + 6h^3p_1(x_N)]w_{N+1} + [60 - 24hp_3(x_N)]hw'_{N+1} = 12h^4r(x_N)$ ….(3.21)

Using equations (3.9), (3.21), and (3.28) together give a system with *N*×*N* seven-diagonal matrix problem

$$Aw = \text{c} \qquad \text{…(3.29)}$$

where

A=

$$\begin{bmatrix} 192-G_1 & -(108+G_2) & 32-G_3 & -3 & & & & & \\ -(8+F_2(x_2)) & 12-F_3(x_2) & -(8+F_4(x_2)) & 2+F_1(x_2) & & & & & \\ 2-F_1(x_3) & -(8+F_2(x_3)) & 12-F_3(x_3) & -(8+F_4(x_3)) & 2+F_1(x_3) & & & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & & & \\ & & & & & 2-F_1(x_{N-1}) & -(8+F_2(x_{N-1})) & 12-F_3(x_{N-1}) & -(8+F_4(x_{N-1})) \\ & & & & & -3 & 32-G_{N2} & -(108+G_{N1}) & 192-G_N \end{bmatrix}_{N\times N}$$

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{N-1} \\ w_N \end{bmatrix}_{N\times 1}, \quad c = \begin{bmatrix} (113+G_0)\alpha_0 + (60+24hp_3(x_1))h^2 + 12h^4r(x_1) \\ (-2+F_1(x_2))\alpha_0 + 2h^4r(x_2) \\ 2h^4r(x_3) \\ \vdots \\ 2h^4r(x_{N-2}) \\ -(2+F_1(x_{N-1}))\beta_0 + 2h^4r(x_{N-1}) \\ (113+G_{N0})\beta_0 - (60-24hp_3(x_N))h\beta_1 + 12h^4r(x_N) \end{bmatrix}_{N\times 1}$$

$F_1(x_i) = -hp_3(x_i)$, $F_2(x_i) = 2hp_3(x_i) + 2h^2p_2(x_i) - h^3p_1(x_i)$

$F_3(x_i) = -4h^2p_2(x_i) + 2h^4p_0(x_i)$, $F_4(x_i) = -2hp_3(x_i) + 2h^2p_2(x_i) + h^3p_1(x_i)$

$G_0 = 32hp_3(x_1) + 12h^2p_2(x_1) - 6h^3p_1(x_1),$

$G_1 = -36hp_3(x_1) - 24h^2p_2(x_1) + 12h^4p_0(x_1)$

$G_2 = 12h^2p_2(x_1) + 6h^3p_1(x_1), G_3 = 4hp_3(x_1)$

$G_{N2} = -4hp_3(x_N), G_{N1} = 12h^2p_2(x_N) - 6h^3p_1(x_N)$

$G_N = 36hp_3(x_N) - 24h^2p_2(x_N) + 12h^4p_0(x_N)$

$G_{N0} = -32hp_3(x_N) + 12h^2p_2(x_N) + 6h^3p_1(x_N)$

We use in some endpoints third-order convergence finite-difference formula to get good approximations at that points.

Now, we will use the LU-decomposition method in MATLAB for solving the linear system (3.29) at several values of step-size (h), and then we will use Extrapolation method to get more accuracy for the method without more calculations.

Algorithm (3.2): Linear Finite-Difference Method (3.2) for Solving Fourth-Order BVPS Case (II): See full MATLAB program (p.218-223)

To approximate the solution of the boundary value problem

$$y^{(iv)} = p_3(x)y''' + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \quad a \le x \le b$$

with four boundary conditions:

$$y(a) = \alpha_0\,, y'(a) = \alpha_1\,, y(b) = \beta_0\,, y'(b) = \beta_1$$

It is the same as Alg.(3.1) with modifications in A ,C, and G's functions

Example 2: Consider the linear fourth-order boundary value problem

$y^{(iv)}(x) = (1 + 2x^2)y''' - 3xy'' - 5x^2y' + (x^3 + 1)y + e^{2x}(x^4 + 5x^3 + x^2 - 7x - 13)$, on $0 \le x \le 1$

with boundary conditions :

$$y(0) = 1, \quad y(1) = 0, \quad y'(0) = 1 , \quad y'(1) = -e^2$$

So $p_3(x) = (1 + 2x^2)$, $p_2(x) = -3x$, $p_1(x) = -5x^2$, $p_0(x) = x^3 + 1$, and $r(x) = e^{2x}(x^4 + 5x^3 + x^2 - 7x - 13)$ on the interval [*0,1*]

Now by applying algorithm (3.2) to the example we get the following results that approximate solutions at several values of h as listed in Table (3.3)

Table 3.3

X_i	Exact(Y)	w(*h=0.1*)	w(*h*=0.05)	w(*h*=0.025)	w(*h*=0.0125)	Ext3(h=0.1)	Ext4(h=0.1)
0.1	**1.09926248**	**1.09931028**	**1.09927527**	**1.09926570**	**1.09926329**	**1.09926243**	**1.09926248**
0.2	**1.19345976**	**1.19362142**	**1.19350158**	**1.19347023**	**1.19346238**	**1.19345966**	**1.19345976**
0.3	**1.27548316**	**1.27577197**	**1.27555712**	**1.27550166**	**1.27548778**	**1.27548302**	**1.27548316**
0.4	**1.33532456**	**1.33571058**	**1.33542305**	**1.33534918**	**1.33533071**	**1.33532438**	**1.33532455**
0.5	**1.35914091**	**1.35956443**	**1.35924894**	**1.35916791**	**1.35914766**	**1.35914072**	**1.35914091**
0.6	**1.32804677**	**1.32843558**	**1.32814624**	**1.32807163**	**1.32805298**	**1.32804656**	**1.32804677**
0.7	**1.21655999**	**1.21685013**	**1.21663488**	**1.21657872**	**1.21656467**	**1.21655980**	**1.21655999**
0.8	**0.99060648**	**0.99076439**	**0.99064822**	**0.99061695**	**0.99060910**	**0.99060633**	**0.99060648**
0.9	**0.60496475**	**0.60500667**	**0.60497691**	**0.60496782**	**0.60496552**	**0.60496465**	**0.60496475**

The previous results reveal that as h decreases the accuracy increases. The maximum error for each approximation is as listed in Table (3.4)

Table 3.4

$\|Y - W\|$	Maximum Error
$\|Y - w_i(h = 0.1)\|$	4.2351×10^{-4}
$\|Y - w_i(h = 0.05)\|$	1.0802×10^{-4}
$\|Y - w_i(h = 0.025)\|$	2.7×10^{-5}
$\|Y - w_i(h = 0.0125)\|$	6.7456×10^{-6}
$\|Y - Ext3_i(h = 0.1)\|$	2.0615×10^{-7}
$\|Y - Ext4_i(h = 0.1)\|$	2.7837×10^{-9}

3.3 Finite-Difference Method (3.3) for Solving Fourth-Order Boundary Value Problem:

Consider the linear fourth-order boundary-value problem:

$$y^{(iv)} = p_3(x)y''' + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \quad a \le x \le b$$

…(3.30)

with third especial case of four boundary conditions:

$$y(a) = \alpha_0\,, y'(a) = \alpha_1\,, y''(b) = \beta_2\,, y'''(b) = \beta_3 \qquad \text{…(3.31)}$$

As in previous sections, we select an integer *N*>0 and divide the interval [*a,b*] into (*N*+1) equal subintervals whose endpoints are the mesh points $x_i = a + ih$ for *i*=0,1,2,…,*N+1*, where the step size $h = (b-a)/(N+1)$.

Second, because of symmetry of the fourth-order derivate we use the finite difference formula Eq.(3.9) in pervious sections for i=2,3,…N-2,

$$[2 + hp_3(x_i)]w_{i-2} - [8 + 2hp_3(x_i) + 2h^2p_2(x_i) - h^3p_1(x_i)]w_{i-1} + [12 + 4h^2p_2(x_i) + h^4p_0(x_i)]w_i - [8 - 2hp_3(x_i) + 2h^2p_2(x_i) + h^3p_1(x_i)]w_{i+1} + [2 - hp_3(x_i)]w_{i+2} = 2h^4r(x_i) \qquad \text{…(3.9)}$$

and we use finite-difference formula Eq.(3.28) for *i*=1

$$-[113 + 32hp_3(x_1) + 12h^2p_2(x_1) - 6h^3p_1(x_1)]w_0 + [192 + 36hp_3(x_1) + 24h^2p_2(x_1) - 12h^4p_0(x_1)]w_1 - [108 + 12h^2p_2(x_1) + 6h^3p_1(x_1)]w_2 + [32 - 4hp_3(x_1)]w_3 - 3w_4 - [60 + 24hp_3(x_1)]hw_0' = 12h^4r(x_1) \qquad \text{…(3.28)}$$

those are centered-difference formulas with second order convergence.

Now we derive special finite-difference formula when i=N-1, N using the boundary conditions, so we define finite-difference approximation for derivatives at i=N-1:

$$y'_{N-1} = \frac{y_N - y_{N-2}}{2h} - \frac{h^2}{6} y'''(\xi_{N-1}) \qquad \ldots(3.32)$$

$$y''_{N-1} = \frac{y_{N-2} - 2y_{N-1} + y_N}{h^2} - \frac{h^2}{12} y^{(iv)}(\eta_{N-1}) \qquad \ldots(3.33)$$

$$y'''_{N-1} = \frac{-56_{N-3} + 96y_{N-2} - 24y_{N-1} - 16y_N + 72h^2 y''_{N+1} - 55h^3 y'''_{N+1}}{145h^3} + \frac{151h^3}{145} y^{(vi)}(\zeta_{N-1}) \qquad \ldots(3.34)$$

$$y^{(iv)}_{N-1} = \frac{128y_{N-3} - 468y_{N-2} + 552y_{N-1} - 212y_N + 84h^2 y''_{N+1} - 40h^3 y'_{N+1}}{145h^4} - \frac{131h^2}{1450} y^{(vi)}(\varphi_{N-1}) \quad .. (3.35)$$

By substituting equations (3.32),(3.33),(3.34) and (3.35) in Eq.(3.30) we get

$$\frac{128y_{N-3} - 468y_{N-2} + 552y_{N-1} - 212y_N + 84h^2 y''_{N+1} - 40h^3 y'''_{N+1}}{145h^4} =$$
$$p_3(x_{N-1}) \left[\frac{-56_{N-3} + 96y_{N-2} - 24y_{N-1} - 16y_N + 72h^2 y''_{N+1} - 55h^3 y'''_{N+1}}{145h^3}\right] +$$
$$p_2(x_{N-1}) \left[\frac{y_{N-2} - 2y_{N-1} + y_{N-2}}{h^2}\right] + p_1(x_{N-1}) \left[\frac{y_N - y_{N-2}}{2h}\right] + p_0(x_N) y_{N-1} + r(x_{N-1}) -$$
$$\frac{h^2}{8700}[1450 p_1(x_N - 1) y'''(\xi_{N-1}) + 725 p_2(x_2) y^{(iv)}(\eta_{N-1}) - 1812 h y^{(vi)}(\zeta_{N-1}) -$$
$$786 y^{(vi)}(\varphi_{N-1})] \qquad \ldots(3.36)$$

We use the third-order convergence for some derivatives to get good approximations for the method values. Now by omitting the error term and multiplying the previous equation by $290h^4$ we get the finite-difference formula at i=N-1

$$[256 + 112hp_3(x_{N-1})]w_{N-3} -$$
$$[936 + 192hp_3(x_{N-1}) + 290h^2 p_2(x_{N-1}) - 145h^3 p_1(x_N)]w_{N-2} +$$
$$[1104 + 48hp_3(x_{N-1}) + 580h^2 p_2(x_{N-1}) - 290h^4 p_0(x_{N-1})]w_{N-1} -$$
$$[424 - 32hp_3(x_{N-1}) + 290h^2 p_2(x_{N-1}) + 145 p_1(x_{N-1})]w_N +$$
$$[168 - 144hp_3(x_{N-1})]h^2 w''_{N+1} - [560 - 80hp_3(x_{N-1})]h^3 w'''_{N+1} = 290h^4 r(x_{N-1}) \ldots(3.37)$$

Now we derive appropriate finite-difference formula at i=N using the boundary conditions and finite-difference approximation with more one point to maintain the second order convergence for fourth derivative in Eq.(3.30) as:

$$y'_N = \frac{-6y_{N-1}+6y_N+3h^2y''_{N+1}-4h^3y'''_{N+1}}{6h} + \frac{11h^3}{24}y_N^{(iv)}(\xi_N) \qquad \ldots(3.38)$$

$$y''_N = \frac{6y_{N-2}-12y_{N-1}+6y_N+19h^2y''_{N+1}-13h^3y'''_{N+1}}{25h^2} + \frac{29h^3}{150}y_N^{(v)}(\eta_N) \qquad \ldots(3.39)$$

$$y'''_N = \frac{22y_{N-3}-162y_{N-2}+258y_{N-1}-118y_N+96h^2y''_{N+1}-25h^3y'''_{N+1}}{145h^3} - \frac{121h^3}{2175}y_N^{(vi)}(\zeta_N) \qquad \ldots(3.40)$$

$$y_N^{(iv)} = \frac{28y_{N-3}-48y_{N-2}+12y_{N-1}+8y_N-36h^2y''_{N+1}+100h^3y'''_{N+1}}{145h^4} - \frac{589h^2}{2175}y_N^{(vi)}(\varphi_N) \qquad \ldots(3.41)$$

By substituting equations (3.38), (3.39), (3.40) and (3.41) in Eq.(3.30) we get the equation

$$\frac{28y_{N-3}-48y_{N-2}+12y_{N-1}+8y_N-36h^2y''_{N+1}+100h^3y'''_{N+1}}{145h^4} = p_3(x_N)[\frac{22y_{N-3}-162y_{N-2}+258y_{N-1}-118y_N+96h^2y''_{N+1}-25h^3y'''_{N+1}}{145h^3}] + p_2(x_N)\left[\frac{6y_{N-2}-12y_{N-1}+6y_N+19h^2y''_{N+1}-13h^3y'''_{N+1}}{25h^2}\right] + p_1(x_N)\left[\frac{-6y_{N-1}+6y_N+3h^2y''_{N+1}-4h^3y'''_{N+1}}{6h}\right] + p_0(x_N)y_N + r(x_N) + \frac{h^3}{17400}[7975p_1(x_N)y^{(iv)}(\xi_1) + 3364p_2(x_N)y^{(v)}(\eta_N) - 968p_3(x_N)y^{(vi)}(\zeta_N) + 4712\frac{1}{h}y^{(vi)}(\varphi_N)] \qquad \ldots(3.42)$$

By omitting error term and multiplying Eq.(3.42) by $4350h^4$we get the finite-difference formula at i=N

$[840 - 660hp_3(x_N)]w_{N-3} - [1440 - 4860hp_3(x_N) + 1044h^2p_2(x_N)]w_{N-2} + [360 - 7740hp_3(x_N) + 2088h^2p_2(x_N) + 4350h^3p_1(x_N)]w_{N-1} + [240 +$

$3540hp_3(x_N) - 1044h^2p_2(x_N) - 4350h^3p_1(x_N) - 4350h^4p_0(x_N)]w_N -$
$[1080 + 2880hp_3(x_N) + 3306h^2p_2(x_N) + 2175h^3p_1(x_N)]h^2w''_{N+1} + [3000 -$
$750hp_3(x_N) + 2262h^2p_2(x_N) + 2900h^3p_1(x_N)]h^3w'''_{N+1} = 4350h^4r(x_N)$ ...(3.43)

Using equations (3.9), (3.28), (3.37), and (3.43) together give a system with $N{\times}N$ seven-diagonal matrix problem

$$Aw = c \qquad \text{...(3.44)}$$

where A=

$$\begin{bmatrix} 192-G_1 & -(108+G_2) & 32-G_3 & -3 & & & & \\ -(8+F_2(x_2)) & 12-F_3(x_2) & -(8+F_4(x_2)) & 2+F_1(x_2) & & & & \\ 2-F_1(x_3) & -(8+F_2(x_3)) & 12-F_3(x_3) & -(8+F_4(x_3)) & 2+F_1(x_3) & & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & & & & 256-G_{N11} & -(936+G_{N12}) & 1104-G_{N13} & -(424+G_{N14}) \\ & & & & 840-G_{N1} & -(1440+G_{N3}) & 360+G_{N2} & -(240+G_{N1}) \end{bmatrix}_{N\times N}$$

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{N-1} \\ w_N \end{bmatrix}_{N\times 1}, \quad c = \begin{bmatrix} (113+G_0)\alpha_0 + (60+24hp_3(x_1))h^2 + 12h^4r(x_1) \\ (-2+F_1(x_2))\alpha_0 + 2h^4r(x_2) \\ 2h^4r(x_3) \\ \vdots \\ 2h^4r(x_{N-2}) \\ -(168-G_{N102})h^2\beta_2 + (80+G_{N103})h^3\beta_3 + 290h^4r(x_{N-1}) \\ (1080+G_{N02})h^2\beta_2 - (3000-G_{N03})h^3\beta_3 + 4350h^4r(x_N) \end{bmatrix}_{N\times 1}$$

$F_1(x_i) = -hp_3(x_i)$, $F_2(x_i) = 2hp_3(x_i) + 2h^2p_2(x_i) - h^3p_1(x_i)$

$F_3(x_i) = -4h^2p_2(x_i) + 2h^4p_0(x_i)$, $F_4(x_i) = -2hp_3(x_i) + 2h^2p_2(x_i) + h^3p_1(x_i)$

$G_0 = 986hp_3(x_1) + 2030h^2p_2(x_1) + 2030h^3p_1(x_1)$,

$G_1 = -3306hp_3(x_1) - 4060h^2p_2(x_1) + 2030h^4p_0(x_1)$

$G_2 = 3654hp_3(x_1) + 2030h^2p_2(x_1) - 1015h^3p_1(x_1)$, $G_3 = -2030hp_3(x_1)$

$G_{N11} = -112hp_3(x_{N-1})$, $G_{N12} = 192hp_3(x_{N-1}) + 290h^2p_2(x_{N-1}) - 145h^3p_1(x_{N-1})$,

$G_{N13} = -48hp_3(x_{N-1}) - 580h^2p_2(x_{N-1}) + 290h^4p_0(x_{N-1})$,

$G_{N14} = -32hp_3(x_{N-1}) + 290h^2p_2(x_{N-1})$

$G_{N102} = 144hp_3(x_{N-1})$, $G_{N103} = -110hp_3(x_{N-1})$

$G_{N1} = 660hp_3(x_N)$, $G_{N2} = -4860hp_3(x_N) + 1044h^2p_2(x_N)$

$G_{N3} = 7740hp_3(x_N) - 2088h^2p_2(x_N) - 4350h^3p_1(x_N)$,

$G_{N4} = -3540hp_3(x_N) + 1044h^2p_2(x_N) + 4350h^3p_1(x_N) + 4350h^4p_0(x_N)$,

$G_{N02} = 2880hp_3(x_N) + 3306h^2p_2(x_N) + 2175h^3p_1(x_N)$

$G_{N03} = -750hp_3(x_N) - 2262h^2p_2(x_N) - 2900h^3p_1(x_N)$

Now we will use the MATLAB LU-decomposition method for solving the linear system (3.44) at several values of step-size (h) and then we will use Extrapolation method to get more accuracy for the method without more calculations.

Algorithm (3.3): Linear Finite-Difference Method (3.3) for Solving Fourth-Order BVPS Case (III): See full MATLAB program (p.223-229)

To approximate the solution of the boundary value problem

$$y^{(iv)} = p_3(x)y''' + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x), \quad a \leq x \leq b$$

with four boundary conditions:

$$y(a) = \alpha_0 \, , y'(a) = \alpha_1 \, , y''(b) = \beta_2 \, , y'''(b) = \beta_1$$

is the same as Algorithm(3.3) with some modification in matrix A and constant vector C and G's functions

Example 3: Consider the linear fourth-order boundary value problem

$y^{(iv)}(x) = (1 + 2x^2)y''' - 3xy'' - 5x^2y' + (x^3 + 1)y + e^{2x}(x^4 + 5x^3 + x^2 - 7x - 13)$, on $0 \le x \le 1$

with boundary conditions :

$$y(0) = 1, \qquad y'(0) = 1\,, \quad y''(1) = -4e^2, \qquad y'''(1) = -12e^2$$

So $p_3(x) = (1 + 2x^2)$, $p_2(x) = -3x$, $p_1(x) = -5x^2$, $p_0(x) = x^3 + 1$, and $r(x) = e^{2x}(x^4 + 5x^3 + x^2 - 7x - 13)$ on the interval $[0, 1]$

Now by applying algorithm (3.3) to the example, we get the following results that approximate solutions at several values of h listed in Table (3.5)

Table 3.5

X_i	Exact(Y)	w(h=0.1)	w(h=0.05)	w(h=0.025)	w(h=0.0125)	Ext3(h=0.1)	Ext4(h=0.1)
0.1	1.09926248	1.10009312	1.09940021	1.09928315	1.09926558	1.09924913	1.09926094
0.2	1.19345976	1.19671377	1.19399729	1.19353953	1.19347144	1.19340662	1.19345360
0.3	1.27548316	1.28263291	1.27666224	1.27565626	1.27550794	1.27536419	1.27546932
0.4	1.33532456	1.34771433	1.33736657	1.33562126	1.33536607	1.33511429	1.33530002
0.5	1.35914091	1.37797513	1.36224660	1.35958788	1.35920205	1.35881483	1.35910273
0.6	1.32804677	1.35437197	1.33239527	1.32866742	1.32812985	1.32758180	1.32799213
0.7	1.21655999	1.25123662	1.22230651	1.21737470	1.21666694	1.21593527	1.21648631
0.8	0.99060648	1.03426402	0.99787739	0.99163254	0.99073897	0.98980442	0.99051151
0.9	0.60496475	0.65792945	0.61384781	0.60621547	0.60512422	0.60397251	0.60484674

The previous results reveal that as h decreases the accuracy increases. The maximum error for each approximation is as listed in Table (3.6)

Table 3.6

$\lvert Y - W \rvert$	Maximum Error
$\lvert Y - w_i(h = 0.1) \rvert$	5.2965×10^{-2}
$\lvert Y - w_i(h = 0.05) \rvert$	8.8831×10^{-3}
$\lvert Y - w_i(h = 0.025) \rvert$	1.2507×10^{-3}
$\lvert Y - w_i(h = 0.0125) \rvert$	1.5947×10^{-4}
$\lvert Y - Ext3_i(h = 0.1) \rvert$	9.9223×10^{-4}
$\lvert Y - Ext4_i(h = 0.1) \rvert$	1.1301×10^{-4}

But we must note that if we use second-order convergence finite-difference formula at end-points, the method gets bad approximations for the solution, so we are concerned with getting higher order of convergence and using all boundary conditions to get better approximation for the problem.

From the previous results that appear in tables (3.2), (3.4), & (3.6) and by comparing them, we conclude that method (3.2) has the best results then method (3.1), finally method (3.3).

Table 3.7

Maximum Error for	Boundary Conditions for Fourth-Order Boundary-Value Problems Case (I)	 Case (II)	 Case (III)
	$y(a)=\alpha_0$ $y''(a)=\alpha_2$ $y(b)=\beta_0$ $y'(b)=\beta_1$	$y(a)=\alpha_0$ $y'(a)=\alpha_1$ $y(b)=\beta_0$ $y'(b)=\beta_1$	$y(a)=\alpha_0$ $y'(a)=\alpha_1$ $y''(b)=\beta_2$ $y'''(b)=\beta_1$
$\boldsymbol{\lvert Y-w_i(h=0.1)\rvert}$	9.9593×10^{-4}	4.2351×10^{-4}	5.2965×10^{-2}
$\boldsymbol{\lvert Y-w_i(h=0.05)\rvert}$	2.5934×10^{-4}	1.0802×10^{-4}	8.8831×10^{-3}
$\boldsymbol{\lvert Y-w_i(h=0.025)\rvert}$	6.5268×10^{-5}	2.7×10^{-5}	1.2507×10^{-3}
$\boldsymbol{\lvert Y-w_i(h=0.0125)\rvert}$	1.6338×10^{-5}	6.7456×10^{-6}	1.5947×10^{-4}
$\boldsymbol{\lvert Y-Ext3_i(h=0.1)\rvert}$	3.1646×10^{-7}	2.0615×10^{-7}	9.9223×10^{-4}
$\boldsymbol{\lvert Y-Ext4_i(h=0.1)\rvert}$	4.5419×10^{-9}	2.7837×10^{-9}	1.1301×10^{-4}

Table (3.7) shows that the bad approximation is in the case (III), where β_0 is absence. And case (II) approximation is better than case (I) approximation, where the derivatives orders in boundary condition in case (II) are first-order and one of derivatives order in case (I) is second-order.

So we conclude that, if the problem has first-order boundary conditions values at each interval endpoints, the approximations for its solution will be better than any other boundary conditions. And the absence of initial value for problem at endpoints interval affect and control the method accuracy.

CHAPTER FOUR

Fifth & Sixth & Seventh-Order Linear Boundary-Value Problems

In this chapter we will discuss the finite-difference methods for solving fifth, sixth and seventh-order linear boundary value problems with few cases for each of them, that are used in several branches in science, see [9].

These methods use large equations with hard work and many permutations using boundary conditions and more (one or two) points to derive suitable form of finite-difference formula for derivatives to maintain the second order convergence for the method.

We will sometimes in this chapter use third-order truncation error for some derivatives less than the order of the problem to get good results for the method.

The results of these methods are not good as we want, for example the Differential Transform Method (DTM) that is used to solve a fifth-order boundary-value problem gets a maximum error 9.94×10^{-7}see[9], which needs hard work using finite-difference method.

4.1 Finite-Difference Method (4.1) For Linear Fifth-Order Boundary-Value Problem:

Consider the linear fifth-order boundary-value problem:

$$y^{(v)}(x) = p_4(x)y^{(iv)} + p_3(x)y'''(x) + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x),$$

$$for\ x, \quad a \le x \le b \qquad \ldots(4.1)$$

With first especial case of five boundary conditions

$$y(a) = \alpha_0\ , y'(a) = \alpha_1\ , y''(a) = \alpha_2\ , y(b) = \beta_0\ , y'(b) = \beta_1 \qquad \ldots(4.2)$$

First, we select an integer N>4 and divide the interval $[a,b]$ into $(N+1)$ equal subintervals whose endpoints are the mesh points $x_i = a + ih$ for i=0,1,2,…,$N+1$, where the step size $h=(b-a)/(N+1)$.

Second, we approximate all derivatives in (4.1) using finite-difference approximation on the interior mesh points.

$$y'(x_i) = \frac{y(x_{i+1}) - y(x_{i-1})}{2h} - \frac{h^2}{6}y'''(\xi_i) \qquad \ldots(4.3)$$

$$y''(x_i) = \frac{y(x_{i-1}) - 2y(x_i) + y(x_{i+1})}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_i) \qquad \ldots(4.4)$$

$$y'''(x_i) = \frac{-y(x_{i-2}) + 2y(x_{i-1}) - 2y(x_{i+1}) + y(x_{i+2})}{2h^3} - \frac{h^2}{4}y^{(v)}(\zeta_i) \qquad \ldots(4.5)$$

$$y^{(iv)}(x_i) = \frac{y(x_{i-2}) - 4y(x_{i-1}) + 6y(x_i) - 4y(x_{i+1}) + y(x_{i+2})}{h^4} - \frac{h^2}{6}y^{(vi)}(\varphi_i) \qquad \ldots(4.6)$$

which are centered-difference formulas with second order convergence. But we take special finite-difference approximation for the fifth-derivative, to be suitable to the problem boundary conditions and maintain the second order convergence.

$$y^{(v)}(x_i) = \frac{y(x_{i-4})-8y(x_{i-3})+25y(x_{i-2})-40y(x_{i-1})+35y(x_i)-16y(x_{i+1})+3y(x_{i+2})}{2h^5} + \frac{h^2}{6}y^{(vii)}(\rho_i)$$

…(4.7)

The substitution of the equations (4.3), (4.4), (4.5), (4.6) and (4.7) in Eq.(4.1) gets the equation

$$\frac{y(x_{i-4})-8y(x_{i-3})+25y(x_{i-2})-40y(x_{i-1})+35y(x_i)-16y(x_{i+1})+3y(x_{i+2})}{2h^5} =$$
$$p_4(x_i)\left[\frac{y(x_{i-2})-4y(x_{i-1})+6y(x_i)-4y(x_{i+1})+y(x_{i+2})}{h^4}\right] +$$
$$p_3(x_i)\left[\frac{-y(x_{i-2})+2y(x_{i-1})-2y(x_{i+1})+y(x_{i+2})}{2h^3}\right] + p_2(x_i)[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}] +$$
$$p_1(x_i)[\frac{y(x_{i+1})-y(x_{i-1})}{2h}] + p_0(x_i)y(x_i) + r(x_i) - \frac{h^2}{12}[2p_1(x_i)y'''(\xi_i) +$$
$$p_2(x_i)y^{(iv)}(\eta_i) + 3p_3(x_i)y^{(v)}(\zeta_i) + 2p_4(x_i)y^{(vi)}(\varphi_i) + 2y^{(vii)}(\rho_i) \quad ...(4.8)$$

A Finite-Difference method with truncation error of order $O(h^2)$ results by using (4.8) together with the boundary conditions (4.2) to define

$$w_0 = \alpha_0 , \quad w'_0 = \alpha_1 , w''_0 = \alpha_2 , w_{N+1} = \beta_0, w'_{N+1} = \beta_1$$

and

$$\frac{w_{i-4}-8w_{i-3}+25w_{i-2}-40w_{i-1}+36w_i-16w_{i+1}+3w_{i+2}}{2h^5} -$$
$$p_4(x_i)[\frac{w_{i-2}-4w_{i-1}+6w_i-4w_{i+1}+w_{i+2}}{h^4}] - p_3(x_i)[\frac{-w_{i-2}+2w_{i-1}-2w_{i+1}+w_{i+2}}{2h^3}] -$$
$$p_2(x_i)[\frac{w_{i-1}-2w_i+w_{i+1}}{h^2}] - p_1(x_i)\left[\frac{w_{i+1}-w_{i-1}}{2h}\right] - p_0(x_i)w_i = r(x_i) \quad …(4.9)$$

For each *i*= 4,5,…,*N-1* .

Now by multiplying Eq.(4.9) by $2h^5$ it can be written as :

$$w_{i-4} - 8w_{i-3} + [25 - 2hp_4(x_i) + h^2p_3(x_i)]w_{i-2} - [40 - 8hp_4(x_i) +$$
$$2h^2p_3(x_i) + 2h^3p_2(x_i) - h^4p_1(x_i)]w_{i-1} + [35 - 12hp_4(x_i) + 4h^3p_2(x_i) -$$
$$2h^5p_0(x_i)]w_i - [16 - 8hp_4(x_i) - 2h^2p_3(x_i) + 2h^3p_2(x_i) + h^4p_1(x_i)]w_{i+1} +$$
$$[3 - 2hp_4(x_i) - h^2p_3(x_i)]w_{i+2} = 2h^5r(x_i) \quad …(4.10)$$

But this formula is suitable for N-4 mesh points, so we need to derive special formula when i=1, 2, 3, and N, using the boundary conditions

$$y_1' = \frac{y_2-y_0}{2h} - \frac{h^2}{6}y'''(\xi_1) \qquad \ldots(4.11)$$

$$y_1'' = \frac{y_0-2y_1+y_2}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_1) \qquad \ldots(4.12)$$

In order to maintain second order convergence we derive new finite-difference approximation using the boundary conditions at i=0 for third, fourth and fifth-order derivatives using algorithm (2.2), which is given by:

$$y_1''' = \frac{39y_0-48y_1+9y_2+30hy_0'+6h^2y_0''}{4h^3} - \frac{h^3}{60}y^{(vi)}(\zeta_1) \qquad \ldots(4.13)$$

$$y_1^{(iv)} = \frac{19y_0-27y_2+8y_3+30hy_0'+18h^2y_0''}{9h^4} - \frac{2h^2}{15}y^{(vi)}(\varphi_1) \qquad \ldots(4.14)$$

$$y_1^{(v)} = \frac{-425y_0+540y_1-135y_2+20y_3-330hy_0'-90h^2y_0''}{9h^5} - \frac{2h^2}{21}y^{(vii)}(\rho_1) \qquad \ldots(4.15)$$

By substituting equations (4.11), (4.12), (4.13), (4.14) and (4.15) in Eq.(4.1) we get

$$\frac{-425y_0+540y_1-135y_2+20y_3-330hy_0'-90h^2y_0''}{9h^5} = p_4(x_1)\left[\frac{19y_0-27y_2+8y_3+30hy_0'+18h^2y_0''}{9h^4}\right] + p_3(x_1)\left[\frac{39y_0-48y_1+9y_2+30hy_0'+6h^2y_0''}{4h^3}\right] + p_2(x_1)\left[\frac{y_0-2y_1+y_2}{h^2}\right] + p_1(x_1)\left[\frac{y_2-y_0}{2h}\right] + p_0(x_1)y(x_1) + r(x_1) - \frac{h^2}{420}[70p_1(x_1)y'''(\xi_1) + 35p_2(x_1)y^{(iv)}(\eta_1) + 7hp_3(x_1)y^{(vi)}(\zeta_1) + 56p_4(x_1)y^{(vi)}(\varphi_1) - 40y^{(vii)}(\rho_1)] \qquad \ldots(4.16)$$

By omitting the error term and multiplying the previous equation by $36h^5$ we get the finite-difference formula at i=1

$-[1700 + 76hp_4(x_1) + 76h^2p_3(x_1) + 36h^3p_2(x_1) - 18h^4p_1(x_1)]w_0 + [2160 + 432h^2p_3(x_1) + 72h^3p_2(x_1) - 36h^5p_0(x_1)]w_1 - [540 - 108hp_4(x_1) + 81h^2p_3(x_1) + 36h^3p_2(x_1) + 18h^4p_1(x_1)]w_2 +$

$[80-32hp_4(x_1)]w_3-[1320+120hp_4(x_1)+270h^2p_3(x_1)]hw_0'-$
$[360+72hp_4(x_1)+54h^2p_3(x_1)]h^2w_0''=36h^5r(x_1)$ ...(4.17)

To find the suitable finite-difference formula at i=2 we follow the same step at i=1

$$y_2'=\frac{y_3-y_1}{2h}-\frac{h^2}{6}y'''(\xi_2) \quad ...(4.18)$$

$$y_2''=\frac{y_1-2y_2+y_3}{h^2}-\frac{h^2}{12}y^{(iv)}(\eta_2) \quad ...(4.19)$$

$$y_2'''=\frac{-y_0+2y_1-2y_3+y_4}{2h^3}-\frac{h^2}{4}y^{(v)}(\zeta_2) \quad ...(4.20)$$

$$y_2^{(iv)}=\frac{y_0-4y_1+6y_2-4y_3+y_4}{h^4}-\frac{h^2}{6}y^{(vi)}(\varphi_2) \quad ...(4.21)$$

$$y_2^{(v)}=\frac{2825y_0-4320y_1+2160y_2-800y_3+135y_4+1860hy_0'+360h^2y_0''}{72h^5}-\frac{h^2}{41}y^{(vii)}(\rho_2)$$

...(4.22)

By substituting equations (4.18), (4.19), (4.20), (4.21) and (4.22) in Eq.(4.1) we get

$$\frac{2825y_0-4320y_1+2160y_2-800y_3+135y_4+1860hy_0'+360h^2y_0''}{72h^5}=$$
$$p_4(x_2)\left[\frac{y_0-4y_1+6y_2-4y_3+y_4}{h^4}\right]+p_3(x_2)\left[\frac{-y_0+2y_1-2y_3+y_4}{2h^3}\right]+p_2(x_2)\left[\frac{y_1-2y_2+y_1}{h^2}\right]+$$
$$p_1(x_2)\left[\frac{y_3-y_1}{2h}\right]+p_0(x_2)y(x_2)+r(x_2)-\frac{h^2}{84}[14p_1(x_2)y'''(\xi_2)+$$
$$7p_2(x_2)y^{(iv)}(\eta_2)+21hp_3(x_2)y^{(v)}(\zeta_2)+14p_4(x_2)y^{(vi)}(\varphi_2)-8y^{(vii)}(\rho_2)]$$

...(4.23)

By omitting the error term and multiplying the previous equation by $72h^5$ we get the finite-difference formula at i=2

$[2825-72hp_4(x_2)+36h^2p_3(x_2)]w_0-[4320-288hp_4(x_2)+$
$72h^2p_3(x_2)+72h^3p_2(x_2)-36h^4p_1(x_2)]w_1+[2160-432hp_4(x_2)+$

$144h^3p_2(x_2) - 72h^5p_0(x_2)]w_2 - [800 - 288hp_4(x_2) - 72h^2p_3(x_2) + 72h^3p_2(x_2) - 36h^4p_1(x_2)]w_3 + [135 - 72hp_4(x_2) - 36h^2p_3(x_2)]w_4 + 1860hw_0' + 360h^2w_0'' = 72h^5r(x_2)$ …(4.24)

To find the suitable finite-difference formula at *i*=3 we follow

$$y_3' = \frac{y_4 - y_2}{2h} - \frac{h^2}{6}y'''(\xi_3) \quad \text{…(4.25)}$$

$$y_3'' = \frac{y_2 - 2y_3 + y_4}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_3) \quad \text{…(4.26)}$$

$$y_3''' = \frac{-y_1 + 2y_2 - 2y_4 + y_5}{2h^3} - \frac{h^2}{4}y^{(v)}(\zeta_3) \quad \text{…(4.27)}$$

$$y_3^{(iv)} = \frac{y_1 - 4y_2 + 6y_3 - 4y_4 + y_5}{h^4} - \frac{h^2}{6}y^{(vi)}(\varphi_3) \quad \text{…(4.28)}$$

$$y_3^{(v)} = \frac{-157y_0 + 400y_1 - 500y_2 + 400y_3 - 175y_4 + 32y_5 - 60hy_0'}{20h^5} + \frac{2h^2}{21}y^{(vii)}(\rho_3) \quad \text{…(4.29)}$$

By substituting equations (4.25), (4.26), (4.27), (4.28) and (4.29) in Eq.(4.1) we get

$\frac{-157y_0+400y_1-500y_2+400y_3-175y_4+32y_5-60hy_0'}{20h^5} = p_4(x_3)\left[\frac{y_1-4y_2+6y_3-4y_4+y_5}{h^4}\right] + p_3(x_3)\left[\frac{-y_1+2y_2-2y_4+y_5}{2h^3}\right] + p_2(x_3)\left[\frac{y_2-2y_3+y_4}{h^2}\right] + p_1(x_3)\left[\frac{y_4-y_2}{2h}\right] + p_0(x_3)y(x_3) + r(x_3) - \frac{h^2}{84}[14p_1(x_3)y'''(\xi_3) + 7p_2(x_3)y^{(iv)}(\eta_3) + 21hp_3(x_3)y^{(v)}(\zeta_3) + 14p_4(x_3)y^{(vi)}(\varphi_3) + 8y^{(vii)}(\rho_3)$ …(4.30)

By omitting the error term and multiplying the previous equation by $20h^5$ we get the finite-difference formula at *i*=3

$-157w_0 + [400 - 20hp_4(x_3) + 10h^2p_3(x_3)]w_1 - [500 - 80hp_4(x_3) + 20h^2p_3(x_3) + 20h^3p_2(x_3) - 10h^4p_1(x_3)]w_2 + [400 - 120hp_4(x_3) + 40h^3p_2(x_3) - 20h^5p_0(x_3)]w_3 - [175 - 80hp_4(x_3) - 20h^2p_3(x_3) + 20h^3p_2(x_3) - 10h^4p_1(x_3)]w_4 + [32 - 20hp_4(x_3) - 10h^2p_3(x_3)]w_5 - 60hw_0' = 20h^5r(x_3)$ …(4.31)

To find the suitable finite-difference formula at *i*=*N* we follow the same step as *i*=1 with more one point for fourth-order derivatives as:

$$y'_N = \frac{y_{N+1}-y_{N-1}}{2h} - \frac{h^2}{6}y'''(\xi_N) \qquad \ldots(4.32)$$

$$y''_N = \frac{y_{N-1}-2y_N+y_{N+1}}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_N) \qquad \ldots(4.33)$$

$$y'''_N = \frac{-y_{N-2}+9y_N-8y_{N+1}+6hy'_0}{3h^3} - \frac{3h^2}{20}y^{(v)}(\zeta_N) \qquad \ldots(4.34)$$

$$y_N^{(iv)} = \frac{-3y_{N-3}+32y_{N-2}-108y_{N-1}+192y_N-113y_{N+1}+60hy'_{N+1}}{12h^4} - \frac{h^3}{12}y^{(viii)}(\varphi_N)$$

$$\ldots(4.35)$$

$$y_N^{(v)} = \frac{16y_{N-4}-125y_{N-3}+400y_{N-2}-700y_{N-1}+800y_N-391y_{N+1}+180hy'_{N+1}}{20h^5} + \frac{8h^2}{21}y^{(vii)}(\rho_N)$$

$$\ldots(4.36)$$

By substituting equations (4.32), (4.33), (4.34), (4.35) and (4.36) in Eq. (4.1) we get

$$\frac{16y_{N-4}-125y_{N-3}+400y_{N-2}-700y_{N-1}+800y_N-391y_{N+1}+180hy'_{N+1}}{20h^5} = p_4(x_N)\left[\frac{-3y_{N-3}+32y_{N-2}-108y_{N-1}+192y_N-113y_{N+1}+60hy'_{N+1}}{12h^4}\right] + p_3(x_N)\left[\frac{-y_{N-2}+9y_N-8y_{N+1}+6hy'_0}{3h^3}\right] + p_2(x_N)\left[\frac{y_{N-1}-2y_N+y_{N+1}}{h^2}\right] + p_1(x_N)\left[\frac{y_{N+1}-y_{N-1}}{2h}\right] + p_0(x_2)y(x_N) + r(x_N) - \frac{h^2}{420}[70p_1(x_N)y'''(\xi_N) + 35p_2(x_N)y^{(iv)}(\eta_N) + 63hp_3(x_N)y^{(v)}(\zeta_N) + 70hp_4(x_N)y^{(viii)}(\varphi_N) + 160y^{(vii)}(\rho_N)] \qquad \ldots(4.37)$$

By omitting the error term and multiplying the previous equation by $20h^5$ we get the finite-difference formula at *i*=N

$48w_{N-4} - [375 - 15hp_4(x_N)]w_{N-3} + [1200 - 160hp_4(x_N) + 20h^2p_3(x_N)]w_{N-2} - [2100 - 540hp_4(x_N) + 60h^3p_2(x_N) - 30h^4p_1(x_N)]w_{N-1} + [2400 - 960hp_4(x_N) -$

$180h^2p_3(x_N) + 120h^3p_2(x_N) - 60h^5p_0(x_N)]w_N - [1173 - 565hp_4(x_N) - 160h^2p_3(x_N) + 60h^3p_2(x_N) + 30h^4p_1(x_N)]w_{N+1} + [540 - 300hp_4(x_N) - 120p_3(x_N)]hw'_{N+1} = 60h^5r(x_N)$ …(4.38)

Using equations (4.9), (4.17), (4.24), (4.31), and (4.37) together give a system with $N{\times}N$ nearly seventh-diagonal matrix problem

$$Aw = c \qquad \text{…(4.39)}$$

Where A=

$$\begin{bmatrix}
(2160 - G_{11}) & -(540 + G_{12}) & (80 - G_{13}) & & & & & 0 \\
-(4320 + 36F_{22}) & (2160 - 36F_{32}) & -(800 + 36F_{42}) & (135 - F_{52}) & & & & \\
(400 - 10F_{13}) & -(500 + 10F_{23}) & (400 - 10F_{33}) & -(175 + 10F_{43}) & (32 - 10F_{53}) & & & \\
-8 & (25 - F_{14}) & -(40 + F_{24}) & (35 - F_{34}) & -(16 + F_{44}) & (3 - F_{54}) & & \\
1 & -8 & (25 - F_{15}) & -(40 + F_{25}) & (35 - F_{35}) & -(16 + F_{45}) & (3 - F_{55}) & \\
\ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\
 & 1 & -8 & (25 - F_{1,N-2}) & -(40 + F_{2,N-2}) & (35 - F_{3,N-2}) & -(16 + F_{4,N-2}) & (3 - F_{5,N-2}) \\
 & & 1 & -8 & (25 - F_{1,N-1}) & -(40 + F_{2,N-1}) & (35 - F_{3,N-1}) & -(16 + F_{4,N-1}) \\
0 & & & 48 & -(375 + G_{N1}) & (1200 - G_{N2}) & -(2100 + G_{N3}) & (2400 - G_{N4})
\end{bmatrix}_{N\times N}$$

$$w = [w_1 \quad w_2 \quad \cdots \quad w_{N-1} \quad w_N]_{1\times N}^{\;T} \;,$$

$$c = \begin{bmatrix}
(1700 + G_{10})\alpha_0 + (1320 + G_{101})h\alpha_1 + (360 + G_{102})h^2\alpha_2 + 36h^5r(x_1) \\
(2825 + G_{20})\alpha_0 - (1860 - G_{201})h\alpha_1 - (360 - G_{202})h^2\alpha_2 + 72h^5r(x_2) \\
-157\alpha_0 + 60h\alpha_1 + 20h^5r(x_3) \\
-\alpha_0 + 2h^5r(x_4) \\
2h^5r(x_5) \\
\vdots \\
2h^5r(x_{N-2}) \\
-3\beta_0 + 20h^5r(x_{N-1}) \\
(1173 + G_{N0})\beta_0 - (540 - G_{N01})h\beta_1 + 60h^5r(x_N)
\end{bmatrix}_{N\times 1}$$

$F_{1i} = 2hp_4(x_i) - h^2p_3(x_i)$, $F_{2i} = -8hp_4(x_i) + 2h^2p_3(x_i) + 2h^3p_2(x_i) - h^4p_1(x_i)$

$F_{3i} = 12hp_4(x_i) - 4h^3p_2(x_i) + 2h^5p_0(x_i)$,

$F_{4i} = -8hp_4(x_i) - 2h^2p_3(x_i) + 2h^3p_2(x_i) + h^4p_1(x_i)$, $F_{5i} = 2hp_4(x_i) + h^2p_3(x_i)$

$G_{11} = -432h^2p_3(x_1) - 72h^4p_2(x_1) + 36h^5p_0(x_1)$,

$G_{12} = -108hp_4(x_1) + 81h^2p_3(x_1) + 36h^3p_2(x_1) + 18h^4p_1(x_1)$,

$G_{13} = 32hp_4(x_1)$, $G_{10} = 76hp_4(x_1) + 351h^2p_3(x_1) + 36h^3p_2(x_1) - 18h^4p_1(x_1)$

$G_{101} = 120hp_4(x_1) + 270h^2p_3(x_1)$, $G_{102} = 72hp_4(x_1) + 54h^2p_3(x_1)$

$G_{N1} = -15hp_4(x_N)$, $\quad G_{N2} = 160hp_4(x_N) - 20h^2p_3(x_N)$,

$G_{N3} = -540hp_4(x_N) + 60h^3p_2(x_N) - 30h^4p_1(x_N)$,

$G_{N4} = 960hp_4(x_N) + 180h^2p_3(x_N) - 120h^3p_2(x_N) + 60h^5p_0(x_N)$,

$G_{N0} = -565hp_4(x_N) - 160h^2p_3(x_N) + 60h^3p_2(x_N) + 30h^4p_1(x_N)$,

$G_{N01} = 300hp_4(x_N) + 120h^2p_3(x_N)$,

Now, we use the MATLAB LU-decomposition method for solving the linear system (4.39) at several values of step-size (h) and then we use Extrapolation method to get more accuracy for the method without more calculations.

Algorithm (4.1): Linear Finite-Difference Method (4.1) For Solving Fifth-Order BVPS Case (I): See full MATLAB program (p.229-235)

To approximate the solution of the boundary value problem

$y^{(v)}(x) = p_4(x)y^{(iv)} + p_3(x)y'''(x) + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x)$,

$for\ x, \quad a \le x \le b$, with five boundary conditions

$y(a) = \alpha_0$, $y'(a) = \alpha_1$, $y''(a) = \alpha_2$, $y(b) = \beta_0$, $y'(b) = \beta_1$

Step(1):Input endpoints a,b; boundary conditions $\alpha_0, \alpha_1, \alpha_2, \beta_0, \beta_1$

Step(2): For *k*=1,2,3,4 (To determine h=0.1,0.05,0.025,0,0125)

- set $N(k)$=$(10*2^{(k-1)})$-1;
- $h=(b-a)/(N_k+1)$;
- do steps(2-9)

Step(3): For i=1,2,3, …,$N(k)$ set $x(i)=a+ih$

Find $p_0(x_i), p_1(x_i), p_2(x_i), p_3(x_i)\ , p_4(x_i) and\ r(x_i)$.

(The values of associated functions on Eq.(4.1)

Step(4): For $i=1,2,3,\ldots,N_k$ set the function's F& G's

The adding of Eq.(4.3),(4.4),(4.5) ,(4.6)&(3.7)finite-difference formula for $y^{(iv)}$, y''' ,y'', y' as F's)

The adding of $y^{(iv)}$,y''' ,y'', y' as Gi's) in Eq.(4.17),(4.24),(4.31) & (4.37)

Step(5): To determine the diagonals for matrix A in the system (4.39) ; For i=1,2,3,…,N_k set

d0=the vector of entries for main diagonal of matrix A

dU1=the vector of entries for the first upper diagonal for A

dU2= the vector of entries for the second upper diagonal for A

dL1= the vector of entries for the first lower diagonal for A

dL2= the vector of entries for the second lower diagonal for A

dL3= the vector of entries for the third lower diagonal for A

C=the constant vector on (c) Eq.(4.39)

Step(6) Factorize A into LU, where L is lower tridiagonal matrix with unit main diagonal and U is upper four-diagonal.

Step(7) For *i=1,2,3,…,N(k)* Solve LY=d to find Y using forward method

Step(8) For $i=N_k-1$ *,..2,1* Solve U*w*=Y to find the approximation solution w using backward method

Step(9) Set N1=9 For $i=1,2,3,\dots N_k$, for j=1,2,4,…,N_k set $W1_{i,k}=w_j$;

Step(10) For *i=1,2,…,N1* set h=(b-a)/N1 ; x(i)=a+ih

Step(11) Extrapolate the solution at h= .1,0.05,0.025,0,0125

Ext2(h=0.1)=(4*W1(h=0.05)-W1(h=0.1))/3;

Ext2(h=0.05)= (4*W1(h=0.025)-W1(h=0.05))/3;

Ext2(h=0.025)= (4*W1(h=0.0125)-W1(h=0.025))/3;

Ext3(h=0.1)=(16*Ext2(h=0.05)-Ext2(h=0.1))/15;

Ext3(h=0.05)=(16*Ext2(h=0.025)-Ext2(h=0.05)/15;

Ext4(h=0.1)=(64*Ext3(h=0.05)-Ext3(h=0.1))/63;

Step(12) Output [x' ,W1',Ext4] (The approximation solutions and Extrapolation)

Example 1: Consider the linear fifth-order boundary value problem

$y^{(v)}(x) = (3x-1)y^{(iv)} - (5x-2)y''' + (3x^3 - 2x^2 + 1)y'' + 2x^2y' - 9x^2y + e^{2x}(12x^4 - 13x^3 + 15x^2 - 56)$, On $0 \leq x \leq 1$

With boundary conditions:

$y(0) = 1, \quad y(1) = 0, \quad y'(0) = 1\,, \quad y'(1) = -e^2\,, y''(0) = 0$

So $p_4(x) = (3x - 1), p_3(x) = -(5x - 2),\ p_2(x) = (3x^3 - 2x^2 + 1)\,,$

$p_1(x) = 2x^2\,, p_0(x) = -9x^2\,,$

And $r(x) = e^{2x}(12x^4 - 13x^3 + 15x^2 - 56)$ on interval $[0,1]$

Now by applying algorithm (4.1) to the example we get the following results that approximated solutions at several values of h as listed in Table (4.1)

Table 4.1

X_i	Exact(Y)	$w(h$=0.1)	$w(h$=0.05)	$w(h$=0.025)	$w(h$=0.0125)	Ext3(h=0.1)	Ext4(h=0.1)
0.1	1.09926248	1.09926201	1.09925931	1.09926110	1.09926213	1.09926192	1.09926253
0.2	1.19345976	1.19353415	1.19345319	1.19345341	1.19345803	1.19345531	1.19346005
0.3	1.27548316	1.27579519	1.27547259	1.27546844	1.27547913	1.27547385	1.27548389
0.4	1.33532456	1.33597214	1.33530363	1.33529983	1.33531780	1.33531308	1.33532567
0.5	1.35914091	1.36005529	1.35910412	1.35910748	1.35913183	1.35913004	1.35914222
0.6	1.32804677	1.32903674	1.32799524	1.32800922	1.32803662	1.32803827	1.32804802
0.7	1.21655999	1.21739837	1.21650386	1.21652551	1.21655071	1.21655453	1.21656097
0.8	0.99060648	0.99112331	0.99056237	0.99058275	0.99060012	0.99060381	0.99060706
0.9	0.60496475	0.60513242	0.60494651	0.60495591	0.60496239	0.60496401	0.60496493

The previous results reveal that as h decreases the accuracy increases. The maximum errors for each approximation are as listed in Table (4.2)

Table 4.2

$\|Y - W\|$	**Maximum Error**
$\|Y - w_i(h = 0.1)\|$	9.8997×10^{-4}
$\|Y - w_i(h = 0.05)\|$	5.6131×10^{-5}
$\|Y - w_i(h = 0.025)\|$	3.7545×10^{-5}
$\|Y - w_i(h = 0.0125)\|$	1.0153×10^{-5}
$\|Y - Ext3_i(h = 0.1)\|$	1.1479×10^{-5}
$\|Y - Ext4_i(h = 0.1)\|$	1.3065×10^{-6}

But we must note that we use third and fourth-order convergence finite-difference formula at end-points in the method to get good approximations for the problem's solutions although the method keeps it general second-order convergence.

4.2 Finite-Difference Method (4.2) For Linear Fifth-Order Boundary-Value Problem:

Consider the linear fifth-order boundary-value problem:

$$y^{(v)}(x) = p_4(x)y^{(iv)} + p_3(x)y'''(x) + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x),$$

$$for\ x, \quad a \le x \le b \qquad \ldots(4.1)$$

With second especial case of five boundary conditions

$$y(a) = \alpha_0\ , y'''(a) = \alpha_3, y(b) = \beta_0\ , y''(b) = \beta_2\ \ , y^{(iv)}(b) = \beta_4 \ldots(4.40)$$

To approximate the solution to this problem we will follow the same steps in the previous sections by dividing the interval [*a,b*] into N+1 equal subintervals whose endpoints $x_i=a+ih$,for $i=0,1,2,\ldots,N+1$, then we substitute all derivatives by appropriate finite-difference formulas.

By using algorithm (1.1) we get a suitable finite-difference approximation for $y^{(v)}$

$$y^{(v)}(x_i) = \frac{-3y(x_{i-2})+16y(x_{i-1})-35y(x_i)+40y(x_{i+1})-25y(x_{i+2})+8y(x_{i+3})-y(x_{i+4})}{2h^5} - \frac{h^2}{6}y^{(vii)}(\rho_i)$$

$$\ldots(4.41)$$

Now we use the equations (4.3), (4.4), (4.5) & (4.6) centered-difference formula for $y', y'', y''', y^{(iv)}$and Eq.(4.41) and substituting in Eq.(4.1) to get the equation

$$\frac{-3y(x_{i-2})+16y(x_{i-1})-35y(x_i)+40y(x_{i+1})-25y(x_{i+2})+8y(x_{i+3})-y(x_{i+4})}{2h^5} =$$
$$p_4(x_i)\left[\frac{y(x_{i-2})-4y(x_{i-1})+6y(x_i)-4y(x_{i+1})+y(x_{i+2})}{h^4}\right] +$$
$$p_3(x_i)\left[\frac{-y(x_{i-2})+2y(x_{i-1})-2y(x_{i+1})+y(x_{i+2})}{2h^3}\right] + p_2(x_i)[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}] +$$

$$p_1(x_i)[\frac{y(x_{i+1})-y(x_{i-1})}{2h}] + p_0(x_i)y(x_i) + r(x_i) - \frac{h^2}{12}[2p_1(x_i)y'''(\xi_i) +$$
$$p_2(x_i)y^{(iv)}(\eta_i) + 3p_3(x_i)y^{(v)}(\zeta_i) + 2p_4(x_i)y^{(vi)}(\varphi_i) - 2y^{(vii)}(\rho_i) \quad \ldots(4.42)$$

A Finite-Difference method with truncation error of order $O(h^2)$ results by using (4.42) together with the boundary conditions (4.40) to define

$$w_0 = \alpha_0\,, \quad w_0''' = \alpha_3\ , w_{N+1} = \beta_0, w_{N+1}'' = \beta_2, w_{N+1}^{(iv)} = \beta_4$$

and

$$\frac{-3w_{i-2}+16w_{i-1}-35w_i+40w_{i+1}-25w_{i+2}+8w_{i+3}-w_{i+4}}{2h^5} -$$
$$p_4(x_i)[\frac{w_{i-2}-4w_{i-1}+6w_i-4w_{i+1}+w_{i+2}}{h^4}] - p_3(x_i)[\frac{-w_{i-2}+2w_{i-1}-2w_{i+1}+w_{i+2}}{2h^3}] -$$
$$p_2(x_i)[\frac{w_{i-1}-2w_i+w_{i+1}}{h^2}] - p_1(x_i)\left[\frac{w_{i+1}-w_{i-1}}{2h}\right] - p_0(x_i)w_i = r(x_i) \quad \ldots(4.43)$$

For each *i*= 2, 3…, *N-3*.

Now by multiplying Eq.(4.43) by $2h^5$ it can be written as :

$$-[3 + 2hp_4(x_i) - h^2p_3(x_i)]w_{i-2} + [16 + 8hp_4(x_i) - 2h^2p_3(x_i) -$$
$$2h^3p_2(x_i) + h^4p_1(x_i)]w_{i-1} - [35 + 12hp_4(x_i) - 4h^3p_2(x_i) + 2h^5p_0(x_i)]w_i +$$
$$[40 + 8hp_4(x_i) + 2h^2p_3(x_i) - 2h^3p_2(x_i) - h^4p_1(x_i)]w_{i+1} - [25 +$$
$$2hp_4(x_i) + h^2p_3(x_i)]w_{i+2} + 8w_{i+3} - w_{i+4} = 2h^5r(x_i) \quad \ldots(4.44)$$

But this formula is suitable for N-4 mesh points, so we need to derive special formula when *i*=1,N-2,N-1,N using the boundary conditions(4.40)

In order to maintain second order convergence we derive new finite-difference approximation using the boundary conditions at *i*=0 for third, fourth and fifth-order derivatives with one more point for the fourth-order derivative using algorithm(1.2) which is given by:

$$y_1''' = \frac{-2y_0+6y_1-6y_2+2y_3+h^3y_0'''}{3h^3} - \frac{h^3}{3}y^{(vi)}(\zeta_1) \quad \ldots(4.45)$$

$$y_1^{(iv)} = \frac{-3y_0+8y_1-6y_2+y_4-4h^3y_0'''}{7h^4} - \frac{5h^2}{21}y^{(vi)}(\varphi_1) \qquad \ldots(4.46)$$

$$y_1^{(v)} = \frac{12y_0-46y_1+68y_2-48y_3+16y_4-2y_5+4h^3y_0'''}{5h^5} + \frac{3h^2}{25}y^{(vii)}(\rho_1) \qquad \ldots(4.47)$$

By substituting equations (4.11), (4.12), the centered finite-difference formula for first and second derivatives and Esq.(4.45) ,(4.46) and (4.47) in Eq.(4.1) we get

$$\frac{12y_0-46y_1+68y_2-48y_3+16y_4-2y_5+4h^3y_0'''}{5h^5} = p_4(x_1)\left[\frac{-3y_0+8y_1-6y_2+y_4-4h^3y_0'''}{7h^4}\right] + p_3(x_1)\left[\frac{-2y_0+6y_1-6y_2+2y_3+h^3y_0'''}{3h^3}\right] + p_2(x_1)\left[\frac{y_0-2y_1+y_2}{h^2}\right] + p_1(x_1)\left[\frac{y_2-y_0}{2h}\right] + p_0(x_1)y(x_1) + r(x_1) - \frac{h^2}{2100}[350p_1(x_1)y'''(\xi_1) + 175p_2(x_1)y^{(iv)}(\eta_1) + 700hp_3(x_1)y^{(vi)}(\zeta_1) + 500p_4(x_1)y^{(vi)}(\varphi_1) + 252y^{(vii)}(\rho_1)] \qquad \ldots(4.48)$$

By omitting the error term and multiplying the previous equation by $210h^5$ get the finite-difference formula at *i*=1

$$[504 + 90hp_4(x_1) + 140h^2p_3(x_1) - 210h^3p_2(x_1) + 105h^4p_1(x_1)]w_0 - [1932 + 240hp_4(x_1) + 420h^2p_3(x_1) - 420h^3p_2(x_1) + 210h^5p_0(x_1)]w_1 + [2856 + 180hp_4(x_1) + 420h^2p_3(x_1) - 210h^3p_2(x_1) - 105h^4p_1(x_1)]w_2 - [2016 + 140h^2p_3(x_1)]w_3 + [672 - 30hp_4(x_1)]w_4 - 84w_5 + [168 + 120hp_4(x_1) - 70h^2p_3(x_1)]h^3w_0''' = 210h^5r(x_1) \qquad \ldots(4.49)$$

To find the suitable finite-difference formula at *i=N-2* we follow the same step as *i*=1:

$$y_{N-2}' = \frac{y_{N-1}-y_{N-3}}{2h} - \frac{h^2}{6}y'''(\xi_{N-2}) \qquad \ldots(4.50)$$

$$y_{N-2}'' = \frac{y_{N-3}-2y_{N-2}+y_{N-1}}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_{N-2}) \qquad \ldots(4.51)$$

$$y_{N-2}''' = \frac{-y_{N-4}+2y_{N-3}-2y_{N-1}+y_N}{2h^3} - \frac{h^2}{4}y^{(v)}(\zeta_{N-2}) \qquad \ldots(4.52)$$

$$y_{N-2}^{(iv)} = \frac{y_{N-4}-4y_{N-3}+6y_{N-2}-4y_{N-1}+y_N}{h^4} - \frac{h^2}{6}y^{(vi)}(\varphi_{N-2}) \qquad \ldots(4.53)$$

$$y_{N-2}^{(v)} = \frac{-23y_{N-4}+118y_{N-3}-242y_{N-2}+248y_{N-1}-127y_N+26y_{N+1}-3h^4y_{N+1}^{(iv)}}{17h^5} + \frac{29h^2}{102}y^{(vii)}(\rho_{N-2})$$

…(4.54)

By substituting equations (4.50), (4.51), (4.52), (4.53) and (4.54) in Eq.(4.1) we get

$$\frac{-23y_{N-4}+118y_{N-3}-242y_{N-2}+248y_{N-1}-127y_N+26y_{N+1}-3h^4y_{N+1}^{(iv)}}{17h^5} = p_4(x_{N-2})\left[\frac{y_{N-4}-4y_{N-3}+6y_{N-2}-4y_{N-1}+y_N}{h^4}\right] + p_3(x_{N-2})\left[\frac{-y_{N-4}+2y_{N-3}-2y_{N-1}+y_N}{2h^3}\right] + p_2(x_{N-2})\left[\frac{y_{N-3}-2y_{N-2}+y_{N-1}}{h^2}\right] + p_1(x_{N-2})\left[\frac{y_{N-1}-y_{N-3}}{2h}\right] + p_0(x_{N-2})y(x_{N-2}) + r(x_{N-2}) - \frac{h^2}{204}[34p_1(x_{N-2})y'''(\xi_{N-2}) + 17p_2(x_{N-2})y^{(iv)}(\eta_{N-2}) + 51p_3(x_{N-2})y^{(v)}(\zeta_{N-2}) + 34p_4(x_{N-2})y^{(vi)}(\varphi_{N-2}) + 59y^{(vii)}(\rho_{N-2})] \quad \text{…(4.55)}$$

By omitting the error term and multiplying the previous equation by $34h^5$ we get the finite-difference formula at *i*=N-2

$$-[46 + 34hp_4(x_{N-2}) - 17h^2p_3(x_{N-2})]w_{N-4} + [236 - 136hp_4(x_{N-2}) - 34h^2p_3(x_{N-2}) - 34h^3p_2(x_{N-2}) + 17h^4p_1(x_{N-2})]w_{N-3} - [484 + 204hp_4(x_{N-2}) - 68h^3p_2(x_{N-2}) - 34h^6p_0(x_{N-2})]w_{N-2} + [496 + 136hp_4(x_{N-2}) + 34h^2p_3(x_{N-2}) - 34h^3p_2(x_{N-2}) - 17h^4p_1(x_{N-2})]w_{N-1} - [254 + 34hp_4(x_{N-2}) + 34h^2p_3(x_{N-2})]w_N + 52w_{N+1} - 6h^4w_{N+1}^{(iv)} = 34h^5r(x_{N-1}) \quad \text{…(4.56)}$$

To derive the finite-difference formula at i=N-1 we will use the centered formulas for first, second, third and fourth-order derivatives and special formula for the fifth-order derivative using the boundary conditions (4.40)

$$y'_{N-1} = \frac{y_N-y_{N-2}}{2h} - \frac{h^2}{6}y'''(\xi_{N-1}) \quad \text{…(4.57)}$$

$$y''_{N-1} = \frac{y_{N-2}-2y_{N-1}+y_N}{h^2} - \frac{h^2}{12}y^{(iv)}(\eta_{N-1}) \quad \text{…(4.58)}$$

$$y'''_{N-1} = \frac{-y_{N-3}+2y_{N-2}-2y_N+y_{N+1}}{2h^3} - \frac{h^2}{4}y^{(v)}(\zeta_{N-1}) \qquad \ldots(4.59)$$

$$y^{(iv)}_{N-1} = \frac{y_{N-3}-4y_{N-2}+6y_{N-1}-4y_N+y_{N+1}}{h^4} - \frac{h^2}{6}y^{(vi)}(\varphi_{N-1}) \qquad \ldots(4.60)$$

$$y^{(v)}_{N-1} = \frac{-481y_{N-3}+2584y_{N-2}-5526y_{N-1}+5224y_N-1801y_{N+1}+660h^2y''_{N+1}-124h^4y^{(iv)}_{N+1}}{302h^5} + \frac{67h^2}{906}y^{(vii)}(\rho_{N-1}) \qquad \ldots(4.61)$$

By substituting equations (4.57), (4.58), (4.59), (4.60) and (4.61) in Eq.(4.1) we get

$$\frac{-481y_{N-3}+2584y_{N-2}-5526y_{N-1}+5224y_N-1801y_{N+1}+660h^2y''_{N+1}-124h^4y^{(iv)}_{N+1}}{302h^5} = p_4(x_{N-1})\left[\frac{y_{N-3}-4y_{N-2}+6y_{N-1}-4y_N+y_{N+1}}{h^4}\right] + p_3(x_{N-1})\left[\frac{-y_{N-3}+2y_{N-2}-2y_N+y_{N+1}}{2h^3}\right] + p_2(x_{N-1})\left[\frac{y_{N-2}-2y_{N-1}+y_N}{h^2}\right] + p_1(x_{N-2})\left[\frac{y_N-y_{N-2}}{2h}\right] + p_0(x_{N-1})y(x_{N-1}) + r(x_{N-1}) - \frac{h^2}{1812}[302p_1(x_{N-1})y'''(\xi_{N-1}) + 151p_2(x_{N-1})y^{(iv)}(\eta_{N-1}) + 453p_3(x_{N-1})y^{(v)}(\zeta_{N-1}) + 302p_4(x_{N-1})y^{(vi)}(\varphi_{N-1}) + 134y^{(vii)}(\rho_{N-1})] \qquad \ldots(4.62)$$

By omitting the error term and multiplying the previous equation by $302h^5$ we get the finite-difference formula at *i*=N-1

$$-[481 + 302hp_4(x_{N-1}) - 151h^2p_3(x_{N-1})]w_{N-3} + [2584 - 1208hp_4(x_{N-1}) - 302h^2p_3(x_{N-1}) - 302h^3p_2(x_{N-1}) + 151h^4p_1(x_{N-1})]w_{N-2} - [5526 + 1812hp_4(x_{N-1}) - 604h^3p_2(x_{N-1}) - 302h^5p_0(x_{N-1})]w_{N-1} + [5224 + 1208hp_4(x_{N-1}) + 302h^2p_3(x_{N-1}) - 302h^3p_2(x_{N-1}) - 151h^4p_1(x_{N-1})]w_N - [1801 + 302hp_4(x_{N-1}) + 151h^2p_3(x_{N-1})]w_{N+1} + 660h^2w''_{N+1} - 124h^4w^{(iv)}_{N+1} = 302h^5r(x_{N-1}) \qquad \ldots(4.63)$$

Finally we will derive the finite-difference formula at i=N by using the centered formulas for first, second-order derivatives Eq.(4.32),(4.33) and special formulas for the third, fourth and fifth-order derivatives using the boundary conditions (4.40) that maintain the second order convergence:

$$y_N''' = \frac{-4y_{N-2}+12y_N-8y_{N+1}+12h^2y''_{N+1}-3h^4y^{(iv)}_{N+1}}{16h^3} + \frac{41h^3}{480}y^{(vi)}(\zeta_N) \qquad \ldots(4.64)$$

$$y_N^{(iv)} = \frac{12y_{N-2}-48y_{N-1}+60y_N-24y_{N+1}+12h^2y''_{N+1}+h^4y^{(iv)}_{N+1}}{12h^4} - \frac{59h^2}{360}y^{(vi)}(\varphi_N) \qquad \ldots(4.65)$$

$$y_N^{(v)} = \frac{-121y_{N-3}+424y_{N-2}-486y_{N-1}+184y_N-y_{N+1}-60h^2y''_{N+1}+176h^4y^{(iv)}_{N+1}}{302h^5} - \frac{106h^2}{453}y^{(vii)}(\rho_N)$$

$$\ldots(4.66)$$

By substituting equations (4.32), (4.33), (4.64), (4.65) and (4.66) in Eq. (4.1) we get

$$\frac{-121y_{N-3}+424y_{N-2}-486y_{N-1}+184y_N-y_{N+1}}{302h^5} =$$
$$p_4(x_N)\left[\frac{12y_{N-2}-48y_{N-1}+60y_N-24y_{N+1}+12h^2y''_{N+1}+h^4y^{(iv)}_{N+1}}{12h^4}\right] +$$
$$p_3(x_N)\left[\frac{-4y_{N-2}+12y_N-8y_{N+1}+12h^2y''_{N+1}-3h^4y^{(iv)}_{N+1}}{16h^3}\right] + p_2(x_N)\left[\frac{y_{N-1}-2y_N+y_{N+1}}{h^2}\right] +$$
$$p_1(x_N)\left[\frac{y_{N+1}-y_{N-1}}{2h}\right] + p_0(x_2)y(x_N) + r(x_N) - \frac{h^2}{217440}[36240p_1(x_N)y'''(\xi_N) +$$
$$18120p_2(x_N)y^{(iv)}(\eta_N) - 18573hp_3(x_N)y^{(vi)}(\zeta_N) + 35636hp_4(x_N)y^{(vi)}(\varphi_N) -$$
$$50880y^{(vii)}(\rho_N)] \qquad \ldots(4.67)$$

By omitting the error term and multiplying the previous equation by $7248h^5$we get the finite-difference formula at *i*=N

$$-2904w_{N-3} + [10176 - 7248hp_4(x_N) + 1812h^2p_3(x_N)]w_{N-2} - [11664 -$$
$$28992hp_4(x_N) + 7248h^3p_2(x_N) - 3624h^4p_1(x_N)]w_{N-1} +$$
$$[4416 - 36240hp_4(x_N) - 5436h^2p_3(x_N) + 14496h^3p_2(x_N) - 7248h^5p_0(x_N)]w_N -$$
$$[24 - 14496hp_4(x_N) - 3624h^2p_3(x_N) + 7248h^3p_2(x_N) + 3624h^4p_1(x_N)]w_{N+1} -$$
$$[1440 + 7248hp_4(x_N) + 5436h^2p_3(x_N)]h^2w''_{N+1} + [4224 - 604hp_4(x_N) +$$
$$1359h^2p_3(x_N)]h^4w^{(iv)}_{N+1} = 7248h^5r(x_N) \qquad \ldots(4.68)$$

Using equations (4.44), (4.49), (4.56), (4.63), and (4.68) together give a system with $N \times N$ nearly seventh-diagonal matrix problem

$$Aw = \mathrm{c} \qquad \ldots(4.69)$$

Where A=

$$\begin{bmatrix}
-(1932+G_{11}) & (2856-G_{12}) & -(2016+G_{13}) & 672-G_{14} & -84 & & & 0 \\
16-F_{21} & -35-F_{31} & 40-F_{41} & -25-F_{51} & 8 & -1 & & \\
-3-F_{11} & 16-F_{22} & -35-F_{32} & 40-F_{42} & -25-F_{52} & 8 & -1 & \\
0 & & & & & & & \\
 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
 & & -3-F_{1N-3} & 16-F_{2N-3} & -35-F_{3N-3} & 40-F_{4N-3} & -25-F_{5N-3} & 8 \\
 & & & -(46-17F_{1,N-2}) & (236-17F_{2,N-2}) & -(484-17F_{3,N-2}) & (496-17F_{4,N-2}) & -(254+17F_{5,N-2}) \\
 & & & & -(481+151F_{1,N-1}) & (2584-151F_{2,N-1}) & -(5526+151F_{3,N-1}) & (5224-151F_{4,N-1}) \\
0 & & & & -(2904+G_{N1}) & (10176-G_{N2}) & -(11664+G_{N3}) & (4416-G_{N4})
\end{bmatrix}_{N\times N}$$

$$w = [w_1 \quad w_2 \quad \cdots \quad w_{N-1} \quad w_N]_{1\times N}{}^{T} \; ,$$

$$c = \begin{bmatrix}
(-504+G_{10})\alpha_0 - (168-G_{103})h^3\alpha_3 + 210h^5 r(x_1) \\
(3+F_{11})\alpha_0 + 2h^5 r(x_2) \\
2h^5 r(x_3) \\
\vdots \\
2h^5 r(x_{N-3}) \\
-52\beta_0 + 6h^4\beta_4 + 34h^5 r(x_{N-2}) \\
(1801+151F_{5,N-1})\beta_0 - 660h^2\beta_2 + 144h^4\beta_4 + 302h^5 r(x_{N-1}) \\
(24+G_{N0})\beta_0 + (1440+G_{N02})h^2\beta_2 - (4224-G_{N04})h^4\beta_4 + 7248h^5 r(x_N)
\end{bmatrix}_{N\times 1}$$

$$F_{1i} = 2hp_4(x_i) - h^2 p_3(x_i) \quad , F_{2i} = -8hp_4(x_i) + 2h^2 p_3(x_i) + 2h^3 p_2(x_i) - h^4 p_1(x_i)$$

$$F_{3i} = 12hp_4(x_i) - 4h^3 p_2(x_i) + 2h^5 p_0(x_i) \, ,$$

$$F_{4i} = -8hp_4(x_i) - 2h^2 p_3(x_i) + 2h^3 p_2(x_i) + h^4 p_1(x_i) \, , F_{5i} = 2hp_4(x_i) + h^2 p_3(x_i)$$

$$G_{11} = 240hp_4(x_1) + 420h^2 p_3(x_1) - 420h^3 p_2(x_1) + 210h^5 p_0(x_1) \, ,$$

$$G_{12} = -180hp_4(x_1) - 420h^2 p_3(x_1) + 210h^3 p_2(x_1) + 105h^4 p_1(x_1) \, ,$$

$G_{13} = 140hp_3(x_1)$, $G_{14} = 30hp_4(x_1)$,$G_{103} = -120hp_4(x_1) + 70h^2p_3(x_1)$,

$G_{N1} = 7248hp_4(x_N) - 1812h^2p_3(x_N)$,

$G_{N2} = -28992hp_4(x_N) + 7248h^3p_2(x_N) - 3624h^4p_1(x_N)$,

$G_{N3} = 36240hp_4(x_N) + 5436h^2p_3(x_N) - 14496h^3p_2(x_N) + 7248h^5p_0(x_N)$,

$G_{N0} = -14496hp_4(x_N) - 3624h^2p_3(x_N) + 7248h^3p_2(x_N) + 3624h^4p_1(x_N)$,

$G_{N02} = 7248hp_4(x_N) + 5436h^2p_3(x_N)$, $G_{N04} = 604hp_4(x_N) - 1359h^2p_3(x_N)$

Now we use the LU-decomposition method in MATLAB for solving the linear system (4.69) at several values of step-size (h) and then we use Extrapolation method to get more accuracy for the method without more calculations.

The Algorithm (4.2) for this method is the same as of Algorithm (4.1) with some differences in the diagonals of matrix A,F& G's functions, and the LU-decomposition procedures suitable. See full program (p.235-241)

Example 2: Consider the linear fifth-order boundary value problem

$y^{(v)}(x) = (3x-1)y^{(iv)} - (5x-2)y''' + (3x^3 - 2x^2 + 1)y'' + 2x^2y' - 9x^2y + e^{2x}(12x^4 - 13x^3 + 15x^2 - 56)$, on $0 \le x \le 1$

With boundary conditions:

$y(0) = 1,\ \ y(1) = 0,\ y'''(0) = -4\ , y''(1) = -4e^2\ , y^{(iv)}(1) = -32e^2$

So $p_4(x) = (3x-1), p_3(x) = -(5x-2),\ p_2(x) = (3x^3 - 2x^2 + 1)$,

$p_1(x) = 2x^2\ , p_0(x) = -9x^2$, And $r(x) = e^{2x}(12x^4 - 13x^3 + 15x^2 - 56)$ on the interval $[0,1]$

Now by applying algorithm (4.2) to the example we get the following results that approximated solutions at several values of h as listed in Table (4.3)

Table 4.3

X_i	Exact(Y)	$w(h=0.1)$	$w(h$=0.05)	$w(h$=0.025)	$w(h$=0.0125)	Ext3(h=0.1)	Ext4(h=0.1)
0.1	1.09926248	1.16159292	1.11970458	1.10487111	1.10072185	1.09953894	1.09929578
0.2	1.19345976	1.30076807	1.22869004	1.20312905	1.19597606	1.19393837	1.19351735
0.3	1.27548316	1.41109850	1.32006109	1.28772282	1.27866887	1.27609194	1.27555635
0.4	1.33532456	1.48365996	1.38414976	1.34873628	1.33881595	1.33599526	1.33540511
0.5	1.35914091	1.50606130	1.40757074	1.37245038	1.36260635	1.35981047	1.35922123
0.6	1.32804677	1.46120644	1.37200634	1.34013377	1.33119455	1.32865869	1.32812007
0.7	1.21655999	1.32570625	1.25264427	1.22648657	1.21914566	1.21706581	1.21662049
0.8	0.99060648	1.06783852	1.01617182	0.99764258	0.99243959	0.99096727	0.99064957
0.9	0.60496475	0.64492951	0.61820497	0.60860998	0.60591457	0.60515264	0.60498715

The previous results reveal that the approximations are not good as h value large or N small, so to get more efficient accuracy, we must use large N with small h, noting that the matrix system A is large with eighth-diagonals, or we can use higher order of extrapolation to get good approximations as listed in Table(4.4)

Table 4.4

$\|Y-W\|$	Maximum Error
$\|Y-w_i(h=0.1)\|$	1.4838×10^{-1}
$\|Y-w_i(h=0.05)\|$	4.8825×10^{-2}
$\|Y-w_i(h=0.025)\|$	1.3412×10^{-2}
$\|Y-w_i(h=0.0125)\|$	3.4914×10^{-3}
$\|Y-Ext3_i(h=0.1)\|$	6.7070×10^{-4}
$\|Y-Ext4_i(h=0.1)\|$	8.0550×10^{-5}

The previous errors table shows that realistic approximations appear when h=0.0125, so this method accuracy must be accelerated in their end points using higher order convergence to get more efficient results or using higher extrapolations that the errors at Ext4(0.1) are small, so we can find Ext4 at several values of h to receive our aim.

By comparing table (4.2) with (4.4) we find that method (4.1) is better than method (4.2), which another evidence that the boundary conditions effect strongly the method accuracy.

4.3 Finite-Difference Method (4.3) For Solving Linear Sixth-Order Boundary-Value Problem:

Consider the linear sixth-order boundary-value problem:

$$y^{(vi)}(x) =$$

$$p_5(x)y^{(v)} + p_4(x)y^{(iv)} + p_3(x)y'''(x) + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x)$$

$$for\, x, \quad a \leq x \leq b \qquad \ldots(4.70)$$

With first especial case of six boundary conditions:

$$y(a) = \alpha_0\ , y'(a) = \alpha_1, y''(a) = \alpha_2\ ,\ y(b) = \beta_0\ , y'(b) = \beta_1\ , y''(b) = \beta_2 \qquad \ldots(4.71)$$

To approximate the solution to this problem we will follow the same steps in the previous sections by dividing the interval [*a,b*] into N+1 equal subintervals (N>5) whose endpoints $x_i=a+ih$,for $i=0,1,2,\ldots,N+1$, then we substitute all derivatives by second order centered finite-difference formulas for each derivatives.

$$y^{(v)}(x_i) = \frac{-y(x_{i-3})+4y(x_{i-2})-5y(x_{i-1})+5y(x_{i+1})-4y(x_{i+2})+y(x_{i+3})}{2h^5} - \frac{h^2}{3}y^{(vii)}(\rho_i)\ .(4.72)$$

$$y^{(vi)}(x_i) = \frac{y(x_{i-3})-6y(x_{i-2})+15y(x_{i-1})-20y(x_i)+15y(x_{i+1})-6y(x_{i+2})+y(x_{i+3})}{h^6} - \frac{h^2}{4}y^{(viii)}(\gamma_i) \qquad \ldots(4.73)$$

Now by substituting the finite-difference formulas for derivatives (4.3),(4.4), (4.5),(4.6) ,(4.72) and (4.73) in Eq.(4.70) we get

$$\frac{y(x_{i-3})-6y(x_{i-2})+15y(x_{i-1})-20y(x_i)+15y(x_{i+1})-6y(x_{i+2})+y(x_{i+3})}{h^6} =$$
$$p_5(x_i)\left[\frac{-y(x_{i-3})+4y(x_{i-2})-5y(x_{i-1})+5y(x_{i+1})-4y(x_{i+2})+y(x_{i+3})}{2h^5}\right]+$$
$$p_4(x_i)\left[\frac{y(x_{i-2})-4y(x_{i-1})+6y(x_i)-4y(x_{i+1})+y(x_{i+2})}{h^4}\right]+$$
$$p_3(x_i)\left[\frac{-y(x_{i-2})+2y(x_{i-1})-2y(x_{i+1})+y(x_{i+2})}{2h^3}\right]+p_2(x_i)\left[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}\right]+$$
$$p_1(x_i)\left[\frac{y(x_{i+1})-y(x_{i-1})}{2h}\right]+p_0(x_i)y(x_i)+r(x_i)-\frac{h^2}{12}[2p_1(x_i)y'''(\xi_i)+p_2(x_i)y^{(iv)}(\eta_i)+$$
$$3p_3(x_i)y^{(v)}(\zeta_i)+2p_4(x_i)y^{(vi)}(\varphi_i)+4p_5(x_i)y^{(vii)}(\rho_i)-3y^{(viii)}(\gamma_i)] \qquad \ldots(4.74)$$

A Finite-Difference method with truncation error of order $O(h^2)$ results by using (4.74) together with the boundary conditions (4.71) to define

$$w_0=\alpha_0\ ,w_0'=\alpha_1,w_0''=\alpha_2\ ,w_{N+1}=\beta_0\ ,w_{N+1}'=\beta_1,w_{N+1}''=\beta_2$$

and

$$\frac{y(x_{i-3})-6y(x_{i-2})+15y(x_{i-1})-20y(x_i)+15y(x_{i+1})-6y(x_{i+2})+y(x_{i+3})}{h^6}-$$
$$p_5(x_i)\left[\frac{-y(x_{i-3})+4y(x_{i-2})-5y(x_{i-1})+5y(x_{i+1})-4y(x_{i+2})+y(x_{i+3})}{2h^5}\right]-$$
$$p_4(x_i)\left[\frac{y(x_{i-2})-4y(x_{i-1})+6y(x_i)-4y(x_{i+1})+y(x_{i+2})}{h^4}\right]-$$
$$p_3(x_i)\left[\frac{-y(x_{i-2})+2y(x_{i-1})-2y(x_{i+1})+y(x_{i+2})}{2h^3}\right]-p_2(x_i)\left[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}\right]-$$
$$p_1(x_i)\left[\frac{y(x_{i+1})-y(x_{i-1})}{2h}\right]-p_0(x_i)y(x_i)=r(x_i) \qquad \ldots(4.75)$$

For each i= 3,4,5,…,N-2 .

Now by multiplying Eq.(4.75) by $2h^6$ it can be written as:

$$[2+hp_5(x_i)]w_{i-3}-[12+4hp_5(x_i)+2h^2p_4(x_i)-h^3p_3(x_i)]w_{i-2}+$$
$$[30+5hp_5(x_i)+8h^2p_4(x_i)-2h^3p_3(x_i)-2h^4p_2(x_i)+h^4p_1(x_i)]w_{i-1}-$$
$$[40+12h^2p_4(x_i)-4h^4p_2(x_i)+2h^6p_0(x_i)]w_i+[30-5hp_5(x_i)+$$

$8h^2p_4(x_i) + 2h^3p_3(x_i) - 2h^4p_2(x_i) - h^5p_1(x_i)]w_{i+1} - [12 - 4hp_5(x_i) + 2h^2p_4(x_i) - h^3p_3(x_i)]w_{i+2} + [2 - hp_5(x_i)]w_{i+3} = 2h^6r(x_i)$ ...(4.76)

But this formula is suitable for N-4 mesh points, so we need to derive special formula when *i*=1, 2, 3, N using the boundary conditions.

We will use equations (4.11),(4.12),(4.13) ,(4.14)and (4.15) in section[4.2] for first up to fourth-order derivatives that are second-order convergence and verify the boundary conditions for problem, but we must derive new finite-difference approximations for sixth-order derivatives that also maintains that conditions as:

$$y_1^{(vi)} = \frac{-749y_0+2754y_1-4320y_2+3820y_3-2025y_4+594y_5-74y_6+180h^2y_0''}{63h^6} + \frac{44h^2}{49}y^{(viii)}(\gamma_1) \quad ...(4.77)$$

Note that we use two more points to derive second-order convergence finite-difference formula for sixth-order derivative with two boundary conditions α_0 and $\alpha_2$instead of three to get smaller truncation error of second-order; that if we use three conditions, the order of error will be one, and if we use α_0 and α_1, we get large value of truncation error $-h^2y^{(viii)}(\gamma_1)$.

Now by substituting equations (4.11), (4.12), (4.13), (4.14), (4.15), and (4.77) in Eq.(4.70) we get

$$\frac{-749y_0+2754y_1-4320y_2+3820y_3-2025y_4+594y_5-74y_6+180h^2y_0''}{63h^6} =$$
$$p_5(x_1)\left[\frac{-425y_0+540y_1-135y_2+20y_3-330hy_0'-90h^2y_0''}{9h^5}\right] +$$
$$p_4(x_1)\left[\frac{19y_0-27y_2+8y_3+30hy_0'+18h^2y_0''}{9h^4}\right] + p_3(x_1)\left[\frac{39y_0-48y_1+9y_2+30hy_0'+6h^2y_0''}{4h^3}\right] +$$
$$p_2(x_1)\left[\frac{y_0-2y_1+y_2}{h^2}\right] + p_1(x_1)\left[\frac{y_2-y_0}{2h}\right] + p_0(x_1)y(x_1) + r(x_1) -$$

$$\frac{h^2}{2940}[490p_1(x_1)y'''(\xi_1) + 245p_2(x_1)y^{(iv)}(\eta_1) + 49hp_3(x_1)y^{(vi)}(\zeta_1) + 392p_4(x_1)y^{(vi)}(\varphi_1) + 280p_5(x_1)y^{(vii)}(\rho_1) - 2640y^{(viii)}(\gamma_1)] \qquad \ldots(4.78)$$

By omitting the error term and multiplying the previous equation by $252h^6$ we get the finite-difference formula at i=1

$$-[2996 - 11900hp_5(x_1) + 532h^2p_4(x_1) + 532h^3p_3(x_1) + 252h^4p_2(x_1) - 126h^5p_1(x_1)]w_0 + [11016 - 15120hp_5(x_1) + 3024h^3p_3(x_1) + 504h^4p_2(x_1) - 252h^6p_0(x_1)]w_1 - [17280 - 3780hp_5(x_1) - 756h^2p_4(x_1) + 567h^3p_3(x_1) + 252h^4p_2(x_1) + 126h^5p_1(x_1)]w_2 + [15280 - 560hp_5(x_1) - 224h^2p_4(x_1)]w_3 - 8100w_4 + 2376w_5 - 296w_6 + [9240hp_5(x_1) - 840h^2p_4(x_1) - 1890h^3p_3(x_1)]hw_0' + [720 + 2520hp_5(x_1) - 504h^2p_4(x_1) - 378h^3p_3(x_1)]h^2w_0'' = 252h^6r(x_1)$$

$$\ldots(4.79)$$

To find the suitable finite-difference formula at i=2, we follow the same steps as i=1.

We will use the equations (4.18),(4.19),(4.20),(4.21) and (4.22) that satisfy the boundary conditions (4.71)and derive a new formula for sixth-order derivatives at i=2 with one more point

$$y_2^{(vi)} = \frac{-969y_0+2160y_1-2250y_2+1600y_3-675y_4+144y_5-10y_6-420hy_0'}{60h^6} - \frac{h^2}{8}y^{(viii)}(\gamma_2)$$

$$\ldots(4.80)$$

By substituting equations (4.18), (4.19), (4.20), (4.21), (4.22) and (4.80) in Eq.(4.70) we get

$$\frac{-969y_0+2160y_1-2250y_2+1600y_3-675y_4+144y_5-10y_6-420hy_0'}{60h^6} = p_5(x_2)\left[\frac{2825y_0-4320y_1+2160y_2-800y_3+135y_4+1860hy_0'+360h^2y_0''}{72h^5}\right] + p_4(x_2)\left[\frac{y_0-4y_1+6y_2-4y_3+y_4}{h^4}\right] + p_3(x_2)\left[\frac{-y_0+2y_1-2y_3+y_4}{2h^3}\right] + p_2(x_2)\left[\frac{y_1-2y_2+y_1}{h^2}\right] + p_1(x_2)\left[\frac{y_3-y_1}{2h}\right] + p_0(x_2)y(x_2) + r(x_2) -$$

$\frac{h^2}{168}[28p_1(x_2)y'''(\xi_2) + 14p_2(x_2)y^{(iv)}(\eta_2) + 42hp_3(x_2)y^{(v)}(\zeta_2) + 28p_4(x_2)y^{(vi)}(\varphi_2) - 4p_5(x_2)y^{(vii)}(\rho_2) + 21y^{(viii)}(\gamma_2)]$ ...(4.81)

By omitting the error term and multiplying the previous equation by $360h^6$ we get the finite-difference formula at i=2

$$-[5814 - 14125hp_5(x_2) + 360h^2p_4(x_2) - 180h^3p_3(x_2)]w_0 + [12960 + 21600hp_5(x_2) + 1440h^2p_4(x_2) - 360h^3p_3(x_2) - 360h^4p_2(x_2) + 180h^5p_1(x_2)]w_1 - [13500 - 10800hp_5(x_2) + 2160h^2p_4(x_2) - 720h^4p_2(x_2) + 360h^6p_0(x_2)]w_2 + [9600 + 4000hp_5(x_2) + 1440h^2p_4(x_2) + 360h^3p_3(x_2) - 360h^4p_2(x_2) + 180h^5p_1(x_2)]w_3 - [4050 + 675hp_5(x_2) + 360h^2p_4(x_2) + 180h^3p_3(x_2)]w_4 + 864w_5 - 60w_6 - [2520 + 9300hp_5(x_2)]hw_0' - [1800hp_5(x_2)]h^2w_0'' = 360h^6r(x_2) \quad \ldots(4.82)$$

Now we will derive the finite-difference formula at i=N-1, by following the same previous steps and using the centered finite-difference approximations for the first up to the fourth-order derivatives which are equations (4.57), (4.58),(4.59) and (4.60) in section[4.3] that satisfy boundary conditions(4.71).

We will derive suitable finite-difference approximations that verify the boundary conditions and maintain the second order convergence using Algorithm(1.2)

$$y_{N-1}^{(v)} = \frac{-135y_{N-3}+800y_{N-2}-2160y_{N-1}+4320y_N-1825y_{N+1}+1860hy'_{N+1}-360h^2y''_{N+1}}{72h^5} - \frac{h^2}{42}y^{(vii)}(\rho_{N-1}) \quad \ldots(4.83)$$

$$y_{N-1}^{(vi)} = \frac{-10y_{N-5}+144y_{N-4}-675y_{N-3}+1600y_{N-2}-2250y_{N-1}+2160y_N-969y_{N+1}+420h^2y''_0}{60h^6} - \frac{h^2}{8}y^{(viii)}(\gamma_{N-1}) \quad \ldots(4.84)$$

by substituting equations (4.57), (4.58), (4.59), (4.60), (4.83) and (4.84) in Eq.(4.70) we get

$$\frac{-10y_{N-5}+144y_{N-4}-675y_{N-3}+1600y_{N-2}-2250y_{N-1}+2160y_N-969y_{N+1}+420hy_0'}{60h^6}=$$
$$p_5(x_{N-1})\left[\frac{-135y_{N-3}+800y_{N-2}-2160y_{N-1}+4320y_N-2825y_{N+1}+1860hy_{N+1}'-360h^2y_{N+1}''}{72h^5}\right]+$$
$$p_4(x_{N-1})\left[\frac{y_{N-3}-4y_{N-2}+6y_{N-1}-4y_N+y_{N+1}}{h^4}\right]+p_3(x_{N-1})\left[\frac{-y_{N-3}+2y_{N-2}-2y_N+y_{N+1}}{2h^3}\right]+$$
$$p_2(x_{N-1})\left[\frac{y_{N-2}-2y_{N-1}+y_N}{h^2}\right]+p_1(x_{N-1})\left[\frac{y_N-y_{N-2}}{2h}\right]+p_0(x_{N-1})y(x_{N-1})+r(x_{N-1})-$$
$$\frac{h^2}{168}[28p_1(x_{N-1})y'''(\xi_{N-1})+14p_2(x_{N-1})y^{(iv)}(\eta_{N-1})+42hp_3(x_{N-1})y^{(vi)}(\zeta_{N-1})+$$
$$28p_4(x_{N-1})y^{(vi)}(\varphi_{N-1})+4p_5(x_{N-1})y^{(vii)}(\rho_{N-1})-21y^{(viii)}(\gamma_{N-1})] \qquad \ldots(4.85)$$

By omitting the error term and multiplying the previous equation by $360h^6$ we get the finite-difference formula at *i=N-1*

$$-60w_{N-5}+864w_{N-4}-[4050-675hp_5(x_{N-1})+360h^2p_4(x_{N-1})-$$
$$180h^3p_3(x_{N-1})]w_{N-1}+[9600-4000hp_5(x_{N-1})+1440h^2p_4(x_{N-1})-$$
$$360h^3p_3(x_2)-360h^4p_2(x_{N-1})+180h^5p_1(x_{N-1})]w_{N-2}-[13500-$$
$$10800hp_5(x_{N-1})+2160h^2p_4(x_{N-1})-720h^4p_2(x_{N-1})+$$
$$360h^6p_0(x_{N-1})]w_{N-1}+[12960-21600hp_5(x_{N-1})+1440h^2p_4(x_{N-1})+$$
$$360h^3p_3(x_{N-1})-360h^4p_2(x_{N-1})+180h^5p_1(x_{N-1})]w_N-[5814-$$
$$1425hp_5(x_{N-1})+360h^2p_4(x_{N-1})+180h^3p_3(x_{N-1})]w_{N+1}+[2520-$$
$$9300hp_5(x_{N-1})]hw_{N+1}'+[1800hp_5(x_{N-1})]h^2w_{N+1}''=360h^6r(x_{N+1})\ldots(4.86)$$

Finally we will derive finite-difference formula at i=N using the boundary conditions (4.71) and the finite difference formula for first and second derivatives Eq.(4.32) and (4.33)in section[4.2] and special formulas for third up to the sixth-order derivatives as:

$$y_N'''=\frac{-9y_{N-1}+48y_N-39y_{N+1}+30hy_{N+1}'-6h^2y_{N+1}''}{4h^3}+\frac{h^3}{60}y^{(vi)}(\zeta_N) \qquad \ldots(4.87)$$

$$y_N^{(iv)}=\frac{8y_{N-2}-27y_{N-1}+19y_{N+1}-30hy_{N+1}'+18h^2y_{N+1}''}{9h^4}-\frac{2h^2}{15}y^{(vi)}(\varphi_N) \qquad \ldots(4.88)$$

$$y_N^{(v)} = \frac{-20y_{N-2}+135y_{N-1}-540y_N+425y_{N+1}-330hy'_{N+1}+90h^2y''_{N+1}}{9h^4} - \frac{2h^2}{21}y^{(vii)}(\rho_N) \quad ..(4.89)$$

$$y_N^{(vi)} = \frac{-74y_{N-5}+594y_{N-4}-2025y_{N-3}+3820y_{N-2}-4320y_{N-1}+2754y_N-749y_{N+1}+180h^2y''_{N+1}}{63h^6} + \frac{44h^2}{99}y^{(viii)}(\gamma_N) \quad \dots(4.90)$$

By substituting equations (4.32), (4.33), (4.87), (4.88), (4.89) and (4.90) in Eq.(4.70) we get

$$\frac{-74y_{N-5}+594y_{N-4}-2025y_{N-3}+3820y_{N-2}-4320y_{N-1}+2754y_N-749y_{N+1}+180h^2y''_{N+1}}{63h^6} = p_5(x_N)[\frac{-20y_{N-2}+135y_{N-1}-540y_N+425y_{N+1}-330hy'_{N+1}+90h^2y''_{N+1}}{9h^4}] - p_4(x_N)\left[\frac{8y_{N-2}-27y_{N-1}+19y_{N+1}-30hy'_{N+1}+18h^2y''_{N+1}}{9h^4}\right] - p_3(x_N)\left[\frac{-9y_{N-1}+48y_N-39y_{N+1}+30hy'_{N+1}-6h^2y''_{N+1}}{4h^3}\right] - p_2(x_N)\left[\frac{y_{N-1}-2y_N+y_{N+1}}{h^2}\right] - p_1(x_N)\left[\frac{y_{N+1}-y_{N-1}}{2h}\right] - p_0(x_N)y(x_N) + r(x_N) - \frac{h^2}{13860}[1155p_1(x_N)y'''(\xi_N) + 2310p_2(x_N)y^{(iv)}(\eta_N) - 231hp_3(x_N)y^{(vi)}(\zeta_N) + 1884hp_4(x_N)y^{(vi)}(\varphi_N) + 1320p_5(x_N)y^{(vii)}(\rho_N) - 6160y^{(viii)}(\gamma_N)] \quad \dots(4.91)$$

By omitting the error term and multiplying the previous equation by $252h^6$ we get the finite-difference formula at *i*=N

$$-296w_{N-5} + 2376w_{N-4} - 8100w_{N-3} + [15280 + 560hp_5(x_N) - 224h^2p_4(x_N)]w_{N-2} - [17280 + 3780hp_5(x_N) - 756h^2p_4(x_N) - 567h^3p_3(x_N) + 252h^4p_2(x_N) - 126h^5p_1(x_N)]w_{N-1} + [11016 + 15120hp_5(x_N) - 3024h^3p_3(x_N) + 504h^4p_2(x_N) - 252h^6p_0(x_N)]w_N - [2996 + 11900hp_5(x_N) + 532h^2p_4(x_N) - 2457h^3p_3(x_N) + 252h^4p_2(x_N) + 126h^5p_1(x_N)]w_{N+1} + [9240hp_5(x_N) + 840h^2p_4(x_N) - 1890h^3p_3(x_N)]hw'_{N+1} + [720 - 2520hp_5(x_N) - 504h^2p_4(x_N) + 378h^3p_3(x_N)]h^2w''_{N+1} = 252h^6r(x_N) \quad \dots(4.92)$$

Using equations (4.76),(4.79),(4.82),(4.86), and (4.92) together give a system with *N*×*N* nearly eleventh-diagonal matrix problem

$$Aw = \mathrm{C} \qquad \ldots(4.93)$$

Where A=

$$\begin{bmatrix}
(11016-G_{11}) & -(17280+G_{12}) & (15280-G_{13}) & -(8100+G_{14}) & 2376 & -296 & & 0 \\
(10800-G_{21}) & -(13500+G_{22}) & (9600-G_{23}) & -(4050+G_{24}) & 864 & -60 & & \\
-(12+F_{23}) & 30-F_{33} & -(40+F_{43}) & 30-F_{53} & -(12+F_{63}) & 2+F_{13} & & \\
2-F_{14} & -(12+F_{24}) & 30-F_{34} & -(40+F_{44}) & 30-F_{54} & -(12+F_{64}) & 2+F_{14} & \\
 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\
 & 2-F_{1N-3} & -(12+F_{2N-3}) & 30-F_{3N-3} & -40-F_{4N-3} & 30-F_{5N-3} & -12-F_{6N-3} & 2+F_{1N-3} \\
 & & 2-F_{1N-2} & -(12+F_{2N-2}) & (30-F_{3N-2}) & -(40+F_{4,N-2}) & (30+F_{5,N-2}) & -(12+F_{6,N-2}) \\
 & & -60 & 864 & -(4050+G_{N11}) & (9600-G_{N12}) & -(13500+G_{N13}) & (12960-G_{N14}) \\
0 & & -296 & 2376 & -(8100+G_{N1}) & (15280-G_{N2}) & -(17280+G_{N3}) & (11016-G_{N4})
\end{bmatrix}_{N\times N}$$

$$w = [w_1 \quad w_2 \qquad \cdots \qquad w_{N-1} \quad w_N]_{1\times N}{}^{T} \quad ,$$

$$c = \begin{bmatrix}
(2996+G_{10})\alpha_0 + G_{101}h\alpha_1 - (720-G_{102})h^2\alpha_2 + 252h^6 r(x_1) \\
(5814+G_{20})\alpha_0 + (2520+G_{201})h\alpha_1 + G_{202}h^2\alpha_2 + 360h^6 r(x_2) \\
(-2+F_{13})\alpha_0 + 2h^6 r(x_3) \\
2h^6 r(x_4) \\
\vdots \\
2h^6 r(x_{N-3}) \\
-(2+F_{1\,N-2})\beta_0 + 2h^6 r(x_{N-2}) \\
(5814+G_{N10})\beta_0 + (2520+G_{N101})h\beta_1 + G_{N102}h^2\beta_2 + 360h^6 r(x_{N-1}) \\
(2996+G_{N0})\beta_0 + G_{N01}h\beta_1 - (720-G_{N02})h^2\beta_2 + 252h^6 r(x_N)
\end{bmatrix}_{N\times 1}$$

$F_{1i} = -hp_5(x_i)$, $F_{2i} = 4hp_5(x_i) + 2h^2p_4(x_i) - h^3p_3(x_i)$,

$F_{3i} = -5hp_5(x_i) - 8h^2p_4(x_i) + 2h^3p_3(x_i) + 2h^4p_2(x_i) - h^5p_1(x_i)$

$F_{4i} = 12h^2p_4(x_i) - 4h^4p_2(x_i) + 2h^6p_0(x_i)$,

$F_{5i} = 5hp_5(x_i) - 8h^2p_4(x_i) - 2h^3p_3(x_i) + 2h^4p_2(x_i) + h^5p_1(x_i)$,

$F_{6i} = -4hp_5(x_i) + 2hp_4(x_i) + h^2p_3(x_i)$,

$G_{10} = -11900hp_5(x_1) + 532h^2p_4(x_1) + 532h^3p_3(x_1) + 252h^4p_2(x_1) - 126h^5p_1(x_1)$

$G_{11} = 15120hp_5(x_1) - 3024h^3p_3(x_1) - 504h^4p_2(x_1) + 252h^6p_0(x_1)$

$G_{12} = -3780hp_5(x_1) - 756h^2p_4(x_1) + 567h^3p_3(x_1) + 252h^4p_2(x_1) +$
$126h^5p_1(x_1)$,

$G_{13} = 560hp_5(x_1) + 1568h^2p_4(x_1)$,

$G_{101} = -9240hp_5(x_1) + 840h^2p_4(x_1) + 1890h^3p_3(x_1)$,

$G_{102} = -2520hp_5(x_1) + 504h^2p_4(x_1) + 378h^3p_3(x_1)$

$G_{20} = -14125hp_5(x_2) + 360h^2p_4(x_2) - 180h^3p_3(x_2)$

$G_{21} = -21600hp_5(x_2) - 1440h^2p_4(x_2) + 360h^3p_3(x_2) + 360h^4p_2(x_2) -$
$180h^5p_1(x_2)$,

$G_{22} = -10800hp_5(x_2) + 2160h^2p_4(x_2) - 720h^4p_2(x_2) + 360h^6p_0(x_2)$,

$G_{23} = -4000hp_5(x_2) - 1440h^2p_4(x_2) - 360h^3p_3(x_2) + 360h^4p_2(x_2) -$
$180h^5p_1(x_2)$,

$G_{24} = 675hp_5(x_2) + 360h^2p_4(x_2) + 180h^3p_3(x_2), G_{201} = 9300hp_5(x_2),$

$G_{202} = 1800hp_5(x_2)$

$G_{N11} = -675hp_5(x_{N-1}) + 360h^2p_4(x_{N-1}) - 180h^3p_3(x_{N-1})$

$G_{N12} = 4000hp_5(x_{N-1}) - 1440h^2p_4(x_{N-1}) + 360h^3p_3(x_{N-1}) + 360h^4p_2(x_{N-1}) +$
$180h^5p_1(x_{N-1})$

$G_{N13} = 10800hp_5(x_{N-1}) + 2160h^2p_4(x_{N-1}) - 720h^4p_2(x_{N-1}) + 360h^6p_0(x_{N-1})$

$G_{N14} = 21600hp_5(x_{N-1}) - 1440h^2p_4(x_{N-1}) - 360h^3p_3(x_{N-1}) + 360h^4p_2(x_{N-1}) -$
$180h^5p_1(x_{N-1})$

$G_{N10} = -9125hp_5(x_{N-1}) + 360h^2p_4(x_{N-1}) + 180h^3p_3(x_{N-1})$

$G_{N101} = 9300hp_5(x_{N-1})$, $G_{N104} = -1800hp_5(x_{N-1})$

$G_{N0} = 11900hp_5(x_N) + 532h^2p_4(x_N) - 2457h^3p_3(x_N) + 252h^4p_2(x_N) + 126h^5p_1(x_N)$

$G_{N1} = -560hp_5(x_N) + 244h^2p_4(x_N)$,

$G_{N2} = 3780hp_5(x_N) - 756h^2p_4(x_N) - 567h^3p_3(x_N) + 252h^4p_2(x_N) - 126h^5p_1(x_N)$

$G_{N3} = -15120hp_5(x_N) + 3024h^3p_3(x_N) - 504h^4p_2(x_N) + 252h^6p_0(x_N)$,

$G_{N01} = -9240p_5(x_N) - 840h^2p_4(x_N) + 1890h^3p_3(x_N)$,

$G_{N02} = 2520hp_5(x_N) + 504h^2p_4(x_N) - 378h^3p_3(x_N)$

Now we use the LU-decomposition method with MATLAB for solving the linear system (4.93) at several values of step-size (h) and then we use Extrapolation method to get more accuracy for the method without more calculations.

Algorithm (4.3): Linear Finite-Difference Method (4.3) For Solving Sixth-Order BVPS Case (I): See full MATLAB program (p.241-252)

To approximate the solution of the boundary value problem

$y^{(vi)}(x) =$

$p_5(x)y^{(v)} + p_4(x)y^{(iv)} + p_3(x)y'''(x) + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x)$

$for\ x, \quad a \le x \le b$, with six boundary conditions

$y(a) = \alpha_0$, $y'(a) = \alpha_1$, $y''(a) = \alpha_2$, $y(b) = \beta_0$, $y'(b) = \beta_1$, $y''(b) = \beta_2$

Step(1):Input endpoints a,b; boundary conditions $\alpha_0, \alpha_1, \alpha_2, \beta_0, \beta_1$

Step(2): For k=2,3,4 (To determine h=0.1,0.05,0.025,0,0125)

- set $N_k=(10*2^{(k-1)})-1$;
- $h=(b-a)/(N_k+1)$;
- do steps(2-9)

Step(3): For i=1,2,3, …,N_k set $x(i)=a+ih$

Find $p_0(x_i), p_1(x_i), p_2(x_i), p_3(x_i)\ , p_4(x_i), p_5(x_i) and\ r(x_i)$.

(The values of associated functions on Eq.(4.70)

Step(4): For $i=1,2,3,\ldots,N_k$ set

d0=the vector of entries for main diagonal of matrix A

dU1=the vector of entries for the first upper diagonal for A

dU2= the vector of entries for the second upper diagonal for A

dU3= the vector of entries for the third upper diagonal for A

dU4= the vector of entries for the fourth upper diagonal for A

dU5= the vector of entries for the fifth upper diagonal for A

dL1= the vector of entries for the first lower diagonal for A

dL2= the vector of entries for the second lower diagonal for A

dL3= the vector of entries for the third lower diagonal for A

dL4= the vector of entries for the fourth lower diagonal for A

dL5= the vector of entries for the fifth lower diagonal for A

C=the constant vector on (c) Eq.(4.93)

Step(6) Factorize A into LU, where L is lower sixth-diagonal matrix with unit main diagonal and U is upper sixth-diagonal.

Step(7) For $i=1,2,3,\ldots,N_k$ Solve LY=d to find Y using forward substitution

Step(8) For $i=N_k-1\ ,..2,1$ Solve Uw=Y to find the approximation solution w using backward substitution

Step(9) Set N1=9 For $i=1,2,3,\ldots N_k$, for j=1,2,4,…,N_k set $W1_{i,k}=w_j$;

Step(10) For $i=1,2,\ldots,N1$ set h=(b-a)/N1 ; x(i)=a+ih

Step(11) Extrapolate the solution at h= .1,0.05,0.025,0,0125

Ext2(h=0.1)=(4*W1(h=0.05)-W1(h=0.1))/3;

Ext2(h=0.05)= (4*W1(h=0.025)-W1(h=0.05))/3;

Ext2(h=0.025)= (4*W1(h=0.0125)-W1(h=0.025))/3;

Ext3(h=0.1)=(16*Ext2(h=0.05)-Ext2(h=0.1))/15;

Ext3(h=0.05)=(16*Ext2(h=0.025)-Ext2(h=0.05)/15;

Ext4(h=0.1)=(64*Ext3(h=0.05)-Ext3(h=0.1))/63;

Step(12) Output [x' ,W1',Ext4] (The approximation solutions and Extrapolation)

Example 3: Consider the linear sixth-order boundary value problem

$y^{(vi)}(x) = 2xy^{(v)} - 3x^3y^{(iv)} - (3x^2 - 1)y''' + (10x + 3)y'' + (3 + 5x - x^2)y' - (2x - x^2)y + e^{2x}(-80x^4 - 37x^3 + 110x^2 - 43x - 121),$

on $0 \le x \le 1$,with boundary conditions:

$y(0) = 1, \quad y(1) = 0, \quad y'(0) = 1,$

$y'(1) = -e^2, y''(0) = 0, y''(1) = -4e^2$

So $p_5(x) = 2x^2$, $p_4(x) = -3x^3$, $p_3(x) = -(3x^2 - 1)$, $p_2(x) = (10x + 3)$,

$p_1(x) = -(3 + 5x - x^2), p_0(x) = (2x - x^2)$, And

$r(x) = e^{2x}(80x^4 + 37x^3 - 110x^2 + 43x + 121)$ on the interval $[0, 1]$

Now by applying algorithm (4.3) to the example we get the following results that approximated solutions at several values of h as listed in Table (4.5)

Table 4.5

X_i	Exact(Y)	w(h=0.05)	w(h=0.025)	w(h=0.0125)	w(h=0.0625)	Ext3(h=0.05)	Ext4(h=0.05)
0.1	1.09926248	1.06088276	1.08968773	1.09688801	1.09867183	1.09928801	1.09926463
0.2	1.19345976	1.12683020	1.17682775	1.18933474	1.19243366	1.19350441	1.19346352
0.3	1.27548316	1.19547087	1.25551029	1.27052987	1.27425108	1.27553726	1.27548772
0.4	1.33532456	1.25690577	1.31575905	1.33047332	1.33411799	1.33537817	1.33532910
0.5	1.35914091	1.29404718	1.34291749	1.35511994	1.35814102	1.35918608	1.35914478
0.6	1.32804677	1.28272323	1.31677271	1.32525438	1.32735260	1.32807887	1.32804957
0.7	1.21655999	1.19149206	1.21034669	1.21502291	1.21617808	1.21657832	1.21656165
0.8	0.99060648	0.98109326	0.98826633	0.99002900	0.99046316	0.99061384	0.99060720
0.9	0.60496475	0.60343798	0.60459807	0.60487495	0.60494253	0.60496607	0.60496490

The previous results reveal that errors are large although h is small, so h must be sufficiently small to get good results or use extrapolation. The errors of approximations are listed in (4.6)

Table 4.6

$\lvert Y-W\rvert$	Maximum Error
$\lvert Y-w_i(h=0.05)\rvert$	8.0012×10^{-2}
$\lvert Y-w_i(h=0.025)\rvert$	1.9973×10^{-2}
$\lvert Y-w_i(h=0.0125)\rvert$	4.9533×10^{-3}
$\lvert Y-w_i(h=0.0625)\rvert$	1.2321×10^{-3}
$\lvert Y-Ext3_i(h=0.05)\rvert$	5.4099×10^{-5}
$\lvert Y-Ext4_i(h=0.05)\rvert$	4.5599×10^{-6}

The previous errors table shows that good approximations appear when h=0.0125, so this method's accuracy must be accelerated at end points using higher order convergence to get more efficient results or using higher extrapolations at which the errors for Ext3, Ext4(0.05) are small, so we can find Ext3, Ext4 at several values of h to receive our aim.

4.4 Finite-Difference Method (4.4) For Solving Linear Sixth-Order Boundary-Value Problem:

Consider the linear sixth-order boundary-value problem:

$y^{(vi)}(x) =$

$p_5(x)y^{(v)} + p_4(x)y^{(iv)} + p_3(x)y'''(x) + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x)$

$for\ x, \quad a \le x \le b$ …(4.70)

With second especial case of six boundary conditions:$y(a) = \alpha_0\ , y'''(a) = \alpha_3, y^{(v)}(a) = \alpha_5\ ,$

$y(b) = \beta_0\ , y''(b) = \beta_2\ , y^{(iv)}(b) = \beta_4$ …(4.94)

To approximate the solution to this problem we will follow the same steps in the previous sections and it's finite-difference formula is similar as for previous problem for i=3,4,…,N-2, so we use Eq.(4.76) in section[4.4], but this formula is suitable for N-4 mesh points, so we need to derive special formula when i=1,2,N-1,N using the boundary conditions (4.94) and Algorithm(1.2)

$$y_1''' = \frac{-2y_0+6y_1-6y_2+2y_3+h^3y_0'''-h^5y_0^{(v)}}{3h^3} - \frac{h^3}{3}y^{(vi)}(\zeta_1) \qquad \ldots(4.95)$$

$$y_1^{(iv)} = \frac{-4y_0+12y_1-12y_2+4y_3-4h^3y_0'''+h^5y_0^{(v)}}{6h^4} - \frac{13h^3}{180}y^{(vi)}(\varphi_1) \qquad \ldots(4.96)$$

$$y_1^{(v)} = \frac{10y_0-36y_1+48y_2-28y_3+6y_4+4h^3y_0'''+3h^5y_0^{(v)}}{10h^5} - \frac{107h^2}{300}y^{(vii)}(\rho_1) \qquad \ldots(4.97)$$

$$y_1^{(vi)} = \frac{-17y_0+90y_1-195y_2+220y_3-135y_4+42y_5-5y_6-12h^5y_0^{(v)}}{25h^6} - \frac{h^2}{20}y^{(viii)}(\gamma_1) \qquad \ldots(4.98)$$

Now by substituting equations (4.11), (4.12), (4.95), (4.96), (4.97) and (4.98) in Eq.(4.70) we get

$$\frac{-17y_0+90y_1-195y_2+220y_3-135y_4+42y_5-5y_6-12h^5y_0^{(v)}}{25h^6} = p_5(x_1)\left[\frac{10y_0-36y_1+48y_2-28y_3+6y_4+4h^3y_0'''+3h^5y_0^{(v)}}{10h^5}\right] + p_4(x_1)\left[\frac{-4y_0+12y_1-12y_2+4y_3-4h^3y_0'''+h^5y_0^{(v)}}{6h^4}\right] + p_3(x_1)\left[\frac{-2y_0+6y_1-6y_2+2y_3+h^3y_0'''-h^5y_0^{(v)}}{3h^3}\right] + p_2(x_1)\left[\frac{y_0-2y_1+y_2}{h^2}\right] + p_1(x_1)\left[\frac{y_2-y_0}{2h}\right] + p_0(x_1)y(x_1) + r(x_1) - \frac{h^2}{900}[75p_1(x_1)y'''(\xi_1) + 150p_2(x_1)y^{(iv)}(\eta_1) + 300hp_3(x_1)y^{(vi)}(\zeta_1) + 5hp_4(x_1)y^{(vi)}(\varphi_1) + 3p_5(x_1)y^{(vii)}(\rho_1) - 45y^{(viii)}(\gamma_1)] \qquad \ldots(4.99)$$

By omitting the error term and multiplying the previous equation by $150h^6$ we get the finite-difference formula at i=1

$-[102 + 150hp_5(x_1) - 100h^2p_4(x_1) - 100h^3p_3(x_1) + 150h^4p_2(x_1) - 75h^5p_1(x_1)]w_0 + [540 + 540hp_5(x_1) - 300h^2p_4(x_1) - 300h^3p_3(x_1) + 300h^4p_2(x_1) - 150h^6p_0(x_1)]w_1 - [1170 + 720hp_5(x_1) - 300h^2p_4(x_1) -$

$300h^3p_3(x_1) + 150h^4p_2(x_1) + 75h^5p_1(x_1)]w_2 + [1320 + 420hp_5(x_1) -$

$100h^2p_4(x_1) - 100h^3p_3(x_1)]w_3 - [810 + 90hp_5(x_1)]w_4 + 252w_5 - 30w_6 -$

$[60hp_5(x_1) - 100h^2p_4(x_1) + 50h^3p_3(x_1)]h^3w_0''' - [72 + 45hp_5(x_1) +$

$25h^2p_4(x_1) - 50h^3p_3(x_1)]h^5w_0^{(v)} = 150h^6r(x_1)$ ...(4.100)

To find the suitable finite-difference formula at *i*=2 we follow the same steps as *i*=1, so we will use the equations (4.18),(4.19),(4.20)and (4.21) that satisfy the boundary conditions (4.94) and derive a new formula for fifth and sixth-order derivatives at i=2 with one more point for sixth-order

$$y_2^{(v)} = \frac{10y_0-36y_1+48y_2-28y_3+6y_4+4h^3y_0'''-2h^5y_0^{(v)}}{5h^5} - \frac{43h^2}{150}y^{(vii)}(\rho_2) \quad ...(4.101)$$

$$y_2^{(vi)} = \frac{4y_0-30y_1+90y_2-140y_3+120y_4-54y_5+10y_6-6h^5y_0^{(v)}}{25h^6} - \frac{13h^2}{20}y^{(viii)}(\gamma_2) \quad ...(4.102)$$

By substituting equations (4.18), (4.19), (4.20), (4.21), (4.100) and (4.101) in Eq.(4.70) we get

$\frac{4y_0-30y_1+90y_2-140y_3+120y_4-54y_5+10y_6-6h^5y_0^{(v)}}{25h^6} =$

$p_5(x_2)\left[\frac{10y_0-36y_1+48y_2-28y_3+6y_4+4h^3y_0'''-2h^5y_0^{(v)}}{5h^5}\right] + p_4(x_2)\left[\frac{y_0-4y_1+6y_2-4y_3+y_4}{h^4}\right] +$

$p_3(x_2)\left[\frac{-y_0+2y_1-2y_3+y_4}{2h^3}\right] + p_2(x_2)\left[\frac{y_1-2y_2+y_3}{h^2}\right] + p_1(x_2)\left[\frac{y_3-y_1}{2h}\right] + p_0(x_2)y(x_2) +$

$r(x_2) - \frac{h^2}{600}[50p_1(x_2)y'''(\xi_2) + 100p_2(x_2)y^{(iv)}(\eta_2) + 150p_3(x_2)y^{(v)}(\zeta_2) +$

$100p_4(x_2)y^{(vi)}(\varphi_2) + 214p_5(x_2)y^{(vii)}(\rho_2) - 30y^{(viii)}(\gamma_2)]$ ...(4.103)

By omitting the error term and multiplying the previous equation by $50h^6$ we get the finite-difference formula at *i*=2

$[8 - 100hp_5(x_2) - 50h^2p_4(x_2) + 25h^3p_3(x_2)]w_0 - [60 - 360hp_5(x_2) -$

$200h^2p_4(x_2) + 50h^3p_3(x_2) + 50h^4p_2(x_2) - 25h^5p_1(x_2)]w_1 +$

$[180 - 480hp_5(x_2) - 300h^2p_4(x_2) + 50h^4p_2(x_2) - 50h^6p_0(x_2)]w_2 - [280 -$

$280hp_5(x_2) - 200h^2p_4(x_2) - 50h^3p_3(x_2) + 100h^4p_2(x_2) + 25h^5p_1(x_2)]w_3 +$

$[240 - 60hp_5(x_2) - 50h^2p_4(x_2) - 25h^3p_3(x_2)]w_4 - 108w_5 + 20w_6 -$
$[40hp_5(x_2)]h^3w_0''' - [12 - 20hp_5(x_2)]h^5w_0^{(v)} = 50h^6r(x_2)$ ...(4.104)

Now we will derive the finite-difference formula at i=N-1, we follow the same previous steps and use the centered finite-difference approximations, for the first up to the fourth-order derivatives which are equations (4.57), (4.58), (4.59) and (4.60), and we will derive suitable finite-difference approximations for sixth-order derivative that verify the boundary conditions (4.94) and maintains the second order convergence using Algorithm(1.2)

$$y_{N-1}^{(vi)} = \frac{-11y_{N-5}+216y_{N-4}-1080y_{N-3}+2560y_{N-2}-3375y_{N-1}+2376y_N-686y_{N+1}+180h^2y_0''}{126h^6} - \frac{69h^2}{392}y^{(viii)}(\gamma_{N-1})$$
...(4.105)

By substituting equations (4.57),(4.58),(4.59),(4.60),(4.61)and(4.105) in Eq.(4.70) we get

$\frac{-11y_{N-5}+216y_{N-4}-1080y_{N-3}+2560y_{N-2}-3375y_{N-1}+2376y_N-686y_{N+1}+180h^2y_0''}{126h^6} =$
$p_5(x_{N-1})\left[\frac{-481y_{N-3}+2584y_{N-2}-5526y_{N-1}+5224y_N-1801y_{N+1}+660h^2y_{N+1}''-124h^4y_{N+1}^{(iv)}}{302h^5}\right] +$
$p_4(x_{N-1})\left[\frac{y_{N-3}-4y_{N-2}+6y_{N-1}-4y_N+y_{N+1}}{h^4}\right] + p_3(x_{N-1})\left[\frac{-y_{N-3}+2y_{N-2}-2y_N+y_{N+1}}{2h^3}\right] +$
$p_2(x_{N-1})\left[\frac{y_{N-2}-2y_{N-1}+y_N}{h^2}\right] + p_1(x_{N-1})\left[\frac{y_N-y_{N-2}}{2h}\right] + p_0(x_{N-1})y(x_{N-1}) + r(x_{N-1}) -$
$\frac{h^2}{177576}[14798p_1(x_{N-1})y'''(\xi_{N-1}) + 29596p_2(x_{N-1})y^{(iv)}(\eta_{N-1}) +$
$44394hp_3(x_{N-1})y^{(vi)}(\zeta_{N-1}) + 29596p_4(x_{N-1})y^{(vi)}(\varphi_{N-1}) -$
$13132p_5(x_{N-1})y^{(vii)}(\rho_{N-1}) - 31257y^{(viii)}(\gamma_{N-1})]$ ...(4.106)

By omitting the error term and multiplying the previous equation by $19026h^6$ we get the finite-difference formula at *i=N-1*

$-1661w_{N-5} + 32616w_{N-4} - [163080 - 30303hp_5(x_{N-1}) + 19026h^2p_4(x_{N-1}) -$
$9513h^3p_3(x_{N-1})]w_{N-3} + [386560 - 162792hp_5(x_{N-1}) + 76104h^2p_4(x_{N-1}) -$

$19026h^3p_3(x_2) - 19026h^4p_2(x_{N-1}) + 9513h^5p_1(x_{N-1})]w_{N-2} - [509625 - 348138hp_5(x_{N-1}) + 114156h^2p_4(x_{N-1}) - 38052h^4p_2(x_{N-1}) + 19026h^6p_0(x_{N-1})]w_{N-1} + [358776 - 329112hp_5(x_{N-1}) + 76104h^2p_4(x_{N-1}) + 19026h^3p_3(x_{N-1}) - 19026h^4p_2(x_{N-1}) - 9513h^5p_1(x_{N-1})]w_N - [103586 - 113463hp_5(x_{N-1}) + 19026h^2p_4(x_{N-1}) + 9513h^3p_3(x_{N-1})]w_{N+1} + [27180 - 41580hp_5(x_{N-1})]h^2w''_{N+1} + [7812hp_5(x_{N-1})]h^4w^{(iv)}_{N+1} = 19026h^6r(x_{N+1})$ ...(4.107)

Finally we will derive finite-difference formula at i=N using equations (4.32), (4.33) for first and second derivatives in section [4.2] and (4.64), (4.65) and (4.66) in section [4.3] for third up to the fifth-order derivatives that satisfied the boundary conditions (4.94).

Now we derive new formula for sixth-order derivative:

$$y_N^{(vi)} = \frac{-13y_{N-5}+102y_{N-4}-327y_{N-3}+548y_{N-2}-507y_{N-1}+246y_N-49y_{N+1}+12h^4y^{(iv)}_{N+1}}{21h^6} + \frac{47h^2}{105}y^{(viii)}(\gamma_N) \quad \text{...(4.108)}$$

By substituting equations (4.32), (4.33), (4.64), (4.65), (4.66) and (4.108) in Eq.(4.70) we get

$$\frac{-13y_{N-5}+102y_{N-4}-327y_{N-3}+548y_{N-2}-507y_{N-1}+246y_N-49y_{N+1}+12h^4y^{(iv)}_{N+1}}{21h^6} = p_5(x_N)[\frac{-121y_{N-3}+424y_{N-2}-486y_{N-1}+184y_N-y_{N+1}-60h^2y''_{N+1}+176h^4y^{(iv)}_{N+1}}{302h^5}] - p_4(x_N)\left[\frac{12y_{N-2}-48y_{N-1}+60y_N-24y_{N+1}+12h^2y''_{N+1}+h^4y^{(iv)}_{N+1}}{12h^4}\right] - p_3(x_N)\left[\frac{-4y_{N-2}+12y_N-8y_{N+1}+12h^2y''_{N+1}-3h^4y^{(iv)}_{N+1}}{16h^3}\right] - p_2(x_N)\left[\frac{y_{N-1}-2y_N+y_{N+1}}{h^2}\right] - p_1(x_N)\left[\frac{y_{N+1}-y_{N-1}}{2h}\right] - p_0(x_N)y(x_N) + r(x_N) - \frac{h^2}{1522080}[126840p_1(x_N)y'''(\xi_N) + 253680p_2(x_N)y^{(iv)}(\eta_N) - 130011hp_3(x_N)y^{(v)}(\zeta_N) + 249452hp_4(x_N)y^{(vi)}(\varphi_N) + 1320p_5(x_N)y^{(vii)}(\rho_N) - 356160y^{(viii)}(\gamma_N)] \quad \text{...(4.109)}$$

By omitting the error term and multiplying the previous equation by $50736h^6$ we get the finite-difference formula at *i*=N

$$-31408w_{N-5} + 246432w_{N-4} - [790032 - 20328hp_5(x_N)]w_{N-3} + [1323968 - 71232p_5(x_N) - 50736h^2p_4(x_N) + 12684h^3p_3(x_N)]w_{N-2} - [1224912 - 85176hp_5(x_N) - 202944h^2p_4(x_N) + 50736h^4p_2(x_N) - 25368h^5p_1(x_N)]w_{N-1} + [594336 + 81648hp_5(x_N) - 2536680h^2p_4(x_N) - 38052h^3p_3(x_N) + 101472h^4p_2(x_N) - 50736h^6p_0(x_N)]w_N - [118384 - 168hp_5(x_N) - 101472h^2p_4(x_N) - 25368h^3p_3(x_N) + 50736h^4p_2(x_N) + 25368h^5p_1(x_N)]w_{N+1} + [10080hp_5(x_N) - 50736h^2p_4(x_N) - 38052h^3p_3(x_N)]h^2w''_{N+1} + [28992 - 29568hp_5(x_N) - 5228h^2p_4(x_N) + 9513h^3p_3(x_N)]h^4w^{(iv)}_{N+1} = 50736h^6r(x_N) \quad \text{...(4.110)}$$

Using equations (4.76),(4.100),(4.104), (4.107), and (4.110) together give a system with *N*×*N* nearly eleventh-diagonal matrix problem

$$Aw = c \qquad \text{...(4.111)}$$

Where *A*=

$$\begin{bmatrix}
(540 - G_{11}) & -(1170 + G_{12}) & (1320 - G_{13}) & -(810 + G_{14}) & 252 & -30 & & 0 \\
-(60 + G_{21}) & 180 - G_{22} & -(280 + G_{23}) & 240 - G_{24} & -(108 - G_{25}) & 20 & & \\
-(12 + F_{23}) & 30 - F_{33} & -(40 + F_{43}) & 30 - F_{53} & -(12 + F_{63}) & 2 + F_{13} & & \\
2 - F_{14} & -(12 + F_{24}) & 30 - F_{34} & -(40 + F_{44}) & 30 - F_{54} & -(12 + F_{64}) & 2 + F_{14} & \\
 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
 & 2 - F_{1N-3} & -(12 + F_{2N-3}) & 30 - F_{3N-3} & -40 - F_{4N-3} & 30 - F_{5N-3} & -12 - F_{6N-3} & 2 + F_{1N-3} \\
 & & 2 - F_{1N-2} & -(12 + F_{2N-2}) & (30 - F_{3N-2}) & -(40 + F_{4,N-2}) & (30 + F_{5,N-2}) & -(12 + F_{6,N-2}) \\
 & & -1661 & 32616 & -(163080 + G_{N11}) & (386560 - G_{N12}) & -(509625 + G_{N13}) & (358776 - G_{N14}) \\
0 & & -31408 & 246432 & -(790032 + G_{N1}) & (1323968 - G_{N2}) & -(1224912 + G_{N3}) & (594336 - G_{N4})
\end{bmatrix}_{N\times N}$$

$$w = [w_1 \quad w_2 \quad \cdots \quad w_{N-1} \quad w_N]_{1\times N}^{T} \quad ,$$

$$c = \begin{bmatrix}
(102 + G_{10})\alpha_0 + G_{103}h^3\alpha_3 + (72 + G_{105})h^5\alpha_5 + 150h^6r(x_1) \\
(-8 + G_{20})\alpha_0 + G_{203}h^3\alpha_3 + (12 + G_{205})h^5\alpha_5 + 50h^6r(x_2) \\
(-2 + F_{13})\alpha_0 + 2h^6r(x_3) \\
2h^6r(x_4) \\
\vdots \\
2h^6r(x_{N-3}) \\
-(2 - F_{1\,N-2})\beta_0 + 2h^6r(x_{N-2}) \\
(103586 + G_{N10})\beta_0 - (27180 - G_{N102})h^2\beta_2 + G_{N104}h^4\beta_4 + 19026h^6r(x_{N-1}) \\
(118384 + G_{N0})\beta_0 + G_{N02}h^2\beta_2 - (28992 - G_{N04})h^4\beta_4 + 50736h^6r(x_N)
\end{bmatrix}_{N\times 1}$$

$F_{1i} = -hp_5(x_i)$, $F_{2i} = 4hp_5(x_i) + 2h^2p_4(x_i) - h^3p_3(x_i)$,

$F_{3i} = -5hp_5(x_i) - 8h^2p_4(x_i) + 2h^3p_3(x_i) + 2h^4p_2(x_i) - h^5p_1(x_i)$

$F_{4i} = 12h^2p_4(x_i) - 4h^4p_2(x_i) + 2h^6p_0(x_i)$,

$F_{5i} = 5hp_5(x_i) - 8h^2p_4(x_i) - 2h^3p_3(x_i) + 2h^4p_2(x_i) + h^5p_1(x_i)$,

$F_{6i} = -4hp_5(x_i) + 2hp_4(x_i) + h^2p_3(x_i)$,

$G_{10} = 150hp_5(x_1) - 100h^2p_4(x_1) - 100h^3p_3(x_1) + 150h^4p_2(x_1) - 75h^5p_1(x_1)$

$G_{11} = -540hp_5(x_1) + 300h^2p_4(x_1) + 300h^3p_3(x_1) - 300h^4p_2(x_1) + 150h^6p_0(x_1)$

$G_{12} = 720hp_5(x_1) - 300h^2p_4(x_1) - 300h^3p_3(x_1) + 150h^4p_2(x_1) + 75h^5p_1(x_1)$,

$G_{13} = -420hp_5(x_1) + 100h^2p_4(x_1) + 100h^3p_3(x_1)$, $G_{14} = 90hp_5(x_1)$,

$G_{103} = 60hp_5(x_1) - 100h^2p_4(x_1) + 50h^3p_3(x_1)$,

$G_{105} = 45hp_5(x_1) + 25h^2p_4(x_1) - 50h^3p_3(x_1)$

$G_{20} = 100hp_5(x_2) + 50h^2p_4(x_2) - 25h^3p_3(x_2)$

$G_{21} = -360hp_5(x_2) - 200h^2p_4(x_2) + 50h^3p_3(x_2) + 50h^4p_2(x_2) - 25h^5p_1(x_2)$,

$G_{22} = 480hp_5(x_2) + 300h^2p_4(x_2) - 100h^4p_2(x_2) + 50h^6p_0(x_2)$,

$G_{23} = -280hp_5(x_2) - 200h^2p_4(x_2) - 50h^3p_3(x_2) + 50h^4p_2(x_2) + 25h^5p_1(x_2)$,

$G_{24} = 60hp_5(x_2) + 50h^2p_4(x_2) + 25h^3p_3(x_2), G_{203} = 40hp_5(x_2),$

$G_{205} = -20hp_5(x_2)$

$G_{N11} = -30303hp_5(x_{N-1}) + 19026h^2p_4(x_{N-1}) - 9513h^3p_3(x_{N-1})$

$G_{N12} =$

$162792hp_5(x_{N-1}) - 76104h^2p_4(x_{N-1}) + 19026h^3p_3(x_{N-1}) + 19026h^4p_2(x_{N-1}) - 9513h^5p_1(x_{N-1})$

$G_{N13} = -348138hp_5(x_{N-1}) + 11415h^2p_4(x_{N-1}) - 38052h^4p_2(x_{N-1}) + 19026h^6p_0(x_{N-1})$

$G_{N14} =$

$329112hp_5(x_{N-1}) - 76104h^2p_4(x_{N-1}) - 9026h^3p_3(x_{N-1}) + 19026h^4p_2(x_{N-1}) + 9513h^5p_1(x_{N-1})$

$G_{N10} = -113463hp_5(x_{N-1}) + 19026h^2p_4(x_{N-1}) + 9513h^3p_3(x_{N-1})$

$G_{N102} = 41580hp_5(x_{N-1})$, $G_{N104} = -7812hp_5(x_{N-1})$

$G_{N0} = -168hp_5(x_N) - 101472h^2p_4(x_N) - 25368h^3p_3(x_N) + 50736h^4p_2(x_N) + 25368h^5p_1(x_N)$

$G_{N1} = -20328hp_5(x_N)$,$G_{N2} = 71235hp_5(x_N) + 50736h^2p_4(x_N) - 12684h^3p_3(x_N)$

$G_{N3} = -81648hp_5(x_N) - 202944h^2p_4(x_N) + 50736h^4p_2(x_N) - 25368h^5p_1(x_N)$,

$G_{N4} = 30912hp_5(x_N) + 253680h^2p_4(x_N) + 38052h^3p_3(x_N) - 101472h^4p_2(x_N) + 50736h^6p_0(x_N)$,

$G_{N02} = -10080hp_5(x_N) + 50736h^2p_4(x_N) + 38052h^3p_3(x_N)$,
$G_{N04} = 29568hp_5(x_N) + 4228h^2p_4(x_N) - 9513h^3p_3(x_N)$

Now we use the LU-decomposition with MATLAB method for solving the linear system (4.111) at several values of step-size (h) and then we use Extrapolation method to get more accuracy for the method without more calculations.

The algorithm (4.4) is as Algorithm (4.3) with some modification in A, c and G's function. See full MATLAB program (p.252-262)

Example 4: Consider the linear sixth-order boundary value problem

$y^{(vi)}(x) = 2xy^{(v)} - 3x^3y^{(iv)} - (3x^2 - 1)y''' + (10x + 3)y'' + (3 + 5x - x^2)y' - (2x - x^2)y + e^{2x}(-80x^4 - 37x^3 + 110x^2 - 43x - 121),$

On $0 \leq x \leq 1$

With boundary conditions:

$y(0) = 1, \quad y'''(0) = -4\,, \ y^{(v)}(0) = -48\,,$

$y(1) = 0 \ \ , y''(1) = -4e^2\,, y^{(iv)}(1) = -32e^2$

So $p_5(x) = 2x^2 \ , p_4(x) = -3x^3,\ p_3(x) = -(3x^2 - 1),\ p_2(x) = (10x + 3)\,,$

$p_1(x) = -(3 + 5x - x^2), p_0(x) = (2x - x^2)$, And

$r(x) = e^{2x}(80x^4 + 37x^3 - 110x^2 + 43x + 121)$ on the interval $[0,1]$

Now by applying algorithm (4.4) to the example we get the following results that approximated solutions at several values of h as listed in Table (4.7)

Table 4.7

X_i	Exact(Y)	w(h=0.05)	w(h=0.025)	w(h=0.0125)	w(h=0.0625)	Ext3(h=0.05)	Ext4(h=0.05)
0.1	1.09926248	1.10573163	1.10171857	1.09994637	1.09963903	1.09928729	1.09955279
0.2	1.19345976	1.20436731	1.19764037	1.19462614	1.19411233	1.19350295	1.19396966
0.3	1.27548316	1.28901484	1.28071442	1.27694528	1.27631290	1.27553832	1.27613905
0.4	1.33532456	1.34988745	1.34099789	1.33691274	1.33623713	1.33538545	1.33605309
0.5	1.35914091	1.37336717	1.36472017	1.36070490	1.36004924	1.35920172	1.35987211
0.6	1.32804677	1.34079590	1.33307461	1.32945779	1.32887359	1.32810227	1.32871690
0.7	1.21655999	1.22691989	1.22066350	1.21771264	1.21724020	1.21660575	1.21711425
0.8	0.99060648	0.99789336	0.99350179	0.99142028	0.99108921	0.99063901	0.99100135
0.9	0.60496475	0.60872251	0.60646055	0.60538534	0.60521502	0.60498162	0.60516995

The previous results reveal that the approximated solutions for first value of h are not good but in the next values of h are some where realstic, but we note at Ext4 column the values start to be far away from the exact result, so the errors at Ext4 are increasing while before that are decreasing, so this means that there are a good range for the h value must be choisen to get good results either using extapolation or any method else. The errors of approximations is listed in (4.8)

Table 4.8

$\lvert Y-W\rvert$	**Maximum Error**
$\lvert Y-w_i(h=0.05)\rvert$	1.4563×10^{-2}
$\lvert Y-w_i(h=0.025)\rvert$	5.6733×10^{-3}
$\lvert Y-w_i(h=0.0125)\rvert$	1.5882×10^{-3}
$\lvert Y-w_i(h=0.0625)\rvert$	9.1258×10^{-4}
$\lvert Y-Ext3_i(h=0.05)\rvert$	6.0891×10^{-5}
$\lvert Y-Ext4_i(h=0.05)\rvert$	7.3120×10^{-4}

The previous errors' table shows that a good approximation appear when h=0.0125, so this method's accuracy must be accelerated in its end points using higher order convergence to get more efficient results.

We can continue in using higher extrapolations until the error for Ext4(0.05) starts to be larger, but any way the approximated solution for problem in this way at h=0.05, 0.0125 is better than that in method(4.3).

4.5 Finite-Difference Method (4.5) For Solving Linear Seventh-Order Boundary-Value Problem:

Consider the linear seventh-order boundary-value problem:

$$y^{(vii)}(x) = p_6(x)y^{(vi)} + p_5(x)y^{(v)} + p_4(x)y^{(iv)} + p_3(x)y'''(x) + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x) \; for\; x, \quad a \le x \le b \qquad \ldots(4.112)$$

With especial seven boundary conditions

$$y(a) = \alpha_0 \;, y'(a) = \alpha_1 \,, y''(a) = \alpha_2, y^{(v)}(a) = \alpha_5 \,,$$

$$y(b) = \beta_0 \;, y'(b) = \beta_4 \quad , y^{(v)}(b) = \beta_5 \qquad \ldots(4.113)$$

To approximate the solution to this problem we will follow the same steps in the previous sections by dividing the interval [*a,b*] into N+1 equal subintervals (N>6) whose endpoints $x_i=a+ih$,for $i=0,1,2,\ldots,N+1$, then we substitute all derivatives by second order centered finite-difference formulas for each derivatives.

$$y^{(vii)}(x_i) = \frac{y_{-5}-10y(x_{i-4})+42y(x_{i-3})-98y(x_{i-2})+140y(x_{i-1})-126y(x_i)+70y(x_{i+1})-22y(x_{i+2})+3y(x_{i+3})}{2h^7} + \frac{h^2}{12}y^{(viiii)}(\omega_i) \qquad \ldots(4.114)$$

We use this especially for two reasons; the first is to decrease number of diagonals in the system matrix, in order to decrease calculations.

The second reason is the non symmetric of the number of the boundary-conditions at the endpoints in the problem, which control the convergence of the method at these points.

Now by substituting the finite-difference formulas for derivatives (4.3),(4.4), (4.5),(4.6),(4.72), (4.73) and (4.114) in Eq.(4.112) we get

$$\frac{y_{-5}-10y(x_{i-4})+42y(x_{i-3})-98y(x_{i-2})+140y(x_{i-1})-126y(x_i)+70y(x_{i+1})-22y(x_{i+2})+3y(x_{i+3})}{2h^7}=$$
$$p_6(x_i)\left[\frac{y(x_{i-3})-6y(x_{i-2})+15y(x_{i-1})-20y(x_i)+15y(x_{i+1})-6y(x_{i+2})+y(x_{i+3})}{h^6}\right]+$$
$$p_5(x_i)\left[\frac{-y(x_{i-3})+4y(x_{i-2})-5y(x_{i-1})+5y(x_{i+1})-4y(x_{i+2})+y(x_{i+3})}{2h^5}\right]+$$
$$p_4(x_i)\left[\frac{y(x_{i-2})-4y(x_{i-1})+6y(x_i)-4y(x_{i+1})+y(x_{i+2})}{h^4}\right]+$$
$$p_3(x_i)\left[\frac{-y(x_{i-2})+2y(x_{i-1})-2y(x_{i+1})+y(x_{i+2})}{2h^3}\right]+p_2(x_i)\left[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}\right]+$$
$$p_1(x_i)\left[\frac{y(x_{i+1})-y(x_{i-1})}{2h}\right]+p_0(x_i)y(x_i)+r(x_i)-\frac{h^2}{12}[p_1(x_i)y'''(\xi_i)+$$
$$2p_2(x_i)y^{(iv)}(\eta_i)+3p_3(x_i)y^{(v)}(\zeta_i)+2p_4(x_i)y^{(vi)}(\varphi_i)+4p_5(x_i)y^{(vii)}(\rho_i)+$$
$$3p_6(x_i)y^{(viii)}(\gamma_i)-5y^{(ix)}(\omega_i)] \qquad \ldots(4.115)$$

A Finite-Difference method with truncation error of order $O(h^2)$ results by using (4.115) together with the boundary conditions (4.113) to define

$$w_0=\alpha_0\ ,w_0'=\alpha_1,w_0'''=\alpha_3\ ,w_0^{(v)}=\alpha_5,$$

$$w_{N+1}=\beta_0\ ,w_{N+1}^{(iv)}=\beta_4,w_{N+1}^{(vi)}=\beta_6$$

and

$$\frac{y_{-5}-10y(x_{i-4})+42y(x_{i-3})-98y(x_{i-2})+140y(x_{i-1})-126y(x_i)+70y(x_{i+1})-22y(x_{i+2})+3y(x_{i+3})}{2h^7}-$$
$$p_6(x_i)\left[\frac{y(x_{i-3})-6y(x_{i-2})+15y(x_{i-1})-20y(x_i)+15y(x_{i+1})-6y(x_{i+2})+y(x_{i+3})}{h^6}\right]-$$
$$p_5(x_i)\left[\frac{-y(x_{i-3})+4y(x_{i-2})-5y(x_{i-1})+5y(x_{i+1})-4y(x_{i+2})+y(x_{i+3})}{2h^5}\right]-$$
$$p_4(x_i)\left[\frac{y(x_{i-2})-4y(x_{i-1})+6y(x_i)-4y(x_{i+1})+y(x_{i+2})}{h^4}\right]-$$
$$p_3(x_i)\left[\frac{-y(x_{i-2})+2y(x_{i-1})-2y(x_{i+1})+y(x_{i+2})}{2h^3}\right]-p_2(x_i)\left[\frac{y(x_{i-1})-2y(x_i)+y(x_{i+1})}{h^2}\right]-$$
$$p_1(x_i)\left[\frac{y(x_{i+1})-y(x_{i-1})}{2h}\right]-p_0(x_i)y(x_i)=r(x_i) \qquad \ldots(4.116)$$

For each i=5,…,N-2 .

Now by multiplying Eq.(4.116) by $2h^6$ it can be written as :

$w_{i-5} - 10w_{i-4} + [42 - 2hp_6(x_i) + h^2p_5(x_i)]w_{i-3} - [98 - 12p_6(x_i) - 4h^2p_5(x_i) - 2h^3p_4(x_i) - h^4p_3(x_i)]w_{i-2} + [140 - 30hp_6(x_i) + 5h^2p_5(x_i) + 8h^3p_4(x_i) - 2h^4p_3(x_i) - 2h^5p_2(x_i) + h^6p_1(x_i)]w_{i-1} - [126 - 40hp_6(x_i) + 12h^3p_4(x_i) - 4h^5p_2(x_i) + 2h^7p_0(x_i)]w_i + [70 - 30hp_6(x_i) - 5h^2p_5(x_i) + 8h^3p_4(x_i) + 2h^4p_3(x_i) - 2h^5p_2(x_i) - h^6p_1(x_i)]w_{i+1} - [22 - 12hp_6(x_i) - 4h^2p_5(x_i) + 2h^3p_4(x_i) - h^4p_3(x_i)]w_{i+2} + [3 - 2hp_6(x_i) - h^2p_5(x_i)]w_{i+3} = 2h^7r(x_i)$ …(4.117)

But this formula is suitable for N-6 mesh points, so we need to derive special formulas when *i*=1, 2, 3, 4, N-1, N using the boundary conditions

To derive new finite-difference formula at i=1 we use equations (4.11) and (4.12) for first and second derivatives and new finite-difference approximations for third up to seventh-order derivatives that satisfy the boundary conditions (4.113) with more one up to three points to maintain second order convergence for formula using Algorithm(1.2).

$$y_1''' = \frac{39y_0 - 48y_1 + 9y_2 + 30hy_0' + 6h^2y_0''}{4h^3} - \frac{h^3}{60}y^{(vi)}(\zeta_1) \quad \text{…(4.13)}$$

$$y_1^{(iv)} = \frac{1135y_0 - 1080y_1 - 135y_2 + 80y_3 - 1110hy_0' + 450h^2y_0'' + 18h^5y_0^{(v)}}{135h^4} - \frac{4h^3}{63}y^{(vi)}(\varphi_1) \quad \text{.(4.118)}$$

$$y_1^{(v)} = \frac{-4375y_0 + 5640y_1 - 1530y_2 + 280y_3 - 15y_4 - 3360hy_0' - 900h^2y_0'' - 8h^5y_0^{(v)}}{70h^5} + \frac{h^3}{1176}y^{(v1ii)}(\rho_1)$$

…(4.119)

$$y_1^{(vi)} = \frac{14875y_0 - 23040y_1 + 11880y_2 - 4480y_3 + 765y_4 + 9660hy_0' + 1800h^2y_0'' - 432h^5y_0^{(v)}}{840h^6} - \frac{79h^2}{392}y^{(viii)}(\gamma_1)$$

…(4.120)

$$y_1^{(vii)} = \frac{-10500y_0 + 23040y_1 - 23535y_2 + 16760y_3 - 7380y_4 + 1800y_5 - 185y_6 - 4620hy_0' + 234h^5y_0^{(v)}}{315h^7} + \frac{369h^2}{1861}y^{(viiii)}(\omega_1) \quad \text{…(4.121)}$$

Now by substituting equations (4.11),(4.12),(4.13) ,(4.118), (4.119), (4.120) and (4.121) in Eq.(4.112) we get

$$\frac{-10500y_0+23040y_1-23535y_2+16760y_3-7380y_4+1800y_5-185y_6-4620hy_0'+234h^5y_0^{(v)}}{315h^7}=$$
$$p_6(x_1)\left[\frac{14875y_0-23040y_1+11880y_2-4480y_3+765y_4+9660hy_0'+1800h^2y_0''-432h^5y_0^{(v)}}{840h^6}\right]+$$
$$p_5(x_1)\left[\frac{-4375y_0+5640y_1-1530y_2+280y_3-15y_4-3360hy_0'-900h^2y_0''-8h^5y_0^{(v)}}{70h^5}\right]+$$
$$p_4(x_1)\left[\frac{1135y_0-1080y_1-135y_2+80y_3-1110hy_0'+450h^2y_0''+18h^5y_0^{(v)}}{135h^4}\right]+$$
$$p_3(x_1)\left[\frac{39y_0-48y_1+9y_2+30hy_0'+6h^2y_0''}{4h^3}\right]+p_2(x_1)\left[\frac{y_0-2y_1+y_2}{h^2}\right]+p_1(x_1)\left[\frac{y_2-y_1}{h}\right]+$$
$$p_0(x_1)y_1+r(x_1)-\frac{h^2}{32828040}[2735670p_1(x_1)y'''(\xi_1)+5471340p_2(x_1)y^{(iv)}(\eta_1)+$$
$$547134hp_3(x_1)y^{(vi)}(\zeta_1)+2084320hp_4(x_1)y^{(vii)}(\varphi_1)-27915hp_5(x_1)y^{(viii)}(\rho_1)+$$
$$6615855p_6(x_1)y^{(vii)}(\gamma_1)+6509160y^{(ix)}(\omega_1)] \qquad ...(4.122)$$

By omitting the error term and multiplying the previous equation by $7560h^6$ we get the finite-difference formula at i=1

$$-[252000+133875hp_6(x_1)-472500h^2p_5(x_1)-63560h^3p_4(x_1)+$$
$$73710h^4p_3(x_1)+7560h^5p_2(x_1)-3780h^6p_1(x_1)]w_0+$$
$$[552960+207360hp_6(x_1)-609120h^2p_5(x_1)+60480h^3p_4(x_1)+$$
$$90720h^4p_3(x_1)+15120h^5p_2(x_1)-7560h^7p_0(x_1)]w_1-$$
$$[564840+106920hp_6(x_1)-165240h^2p_5(x_1)-7560h^3p_4(x_1)+$$
$$17010h^4p_3(x_1)+7560h^5p_2(x_1)+3780h^6p_1(x_1)]w_2+$$
$$[402240+40320hp_6(x_1)-30240h^2p_5(x_1)-4480h^3p_4(x_1)]w_3-[177120+$$
$$6885hp_6(x_1)-1620h^2p_5(x_1)]w_4+43200w_5-4440w_6-$$
$$[4620+86940hp_6(x_1)-362880h^2p_5(x_1)-62160h^3p_4(x_1)+56700-$$
$$30240h^4p_3(x_1)]hw_0'-[16200hp_6(x_1)-97200h^2p_5(x_1)+25200h^3p_4(x_1)+$$
$$11340h^4p_3(x_1)]h^2w_0''+[5616+3888hp_6(x_1)+864h^2p_5(x_1)-$$
$$1008h^3p_4(x_1)]h^5w_0^{(v)}=7560h^7r(x_1) \qquad \ldots(4.123)$$

To find the suitable finite-difference formula at *i*=2 we follow the same steps as *i*=1.

We will use the equations (4.18),(4.19),(4.20) and (4.21) that satisfy the boundary conditions (4.113)and derive a new formula for fifth, sixth and seventh-order derivatives at i=2 with one more point for sixth and seventh-order

$$y_2^{(v)} = \frac{863y_0-1620y_1+1140y_2-440y_3+45y_4+12y_5+420hy_0'-11h^5y_0^{(v)}}{85h^5} - \frac{7h^3}{51}y^{(viii)}(\rho_2) \quad \ldots (4.124)$$

$$y_2^{(vi)} = \frac{-965y_0+2130y_1-2160y_2+1460y_3-555y_4+90y_5-420hy_0'-6h^5y_0^{(v)}}{85h^6} - \frac{19h^2}{68}y^{(viii)}(\gamma_2)$$

…(4.125)

$$\frac{-16415y_0+24192y_1-11340y_2+4480y_3-945y_4+2380y_6-11340hy_0'-2520h^2y_0''}{144h^7} - \frac{17h^2}{72}y^{(ix)}(\omega_2)$$

…(4.126)

By substituting equations (4.18),(4.19),(4.20),(4.21),(4.124),(4.125) and (4.126) in Eq.(4.112) we get

$$\frac{-16415y_0+24192y_1-11340y_2+4480y_3-945y_4+2380y_6-11340hy_0'-2520h^2y_0''}{144h^7} =$$
$$p_6(x_2)\left[\frac{-965y_0+2130y_1-2160y_2+1460y_3-555y_4+90y_5-420hy_0'-6h^5y_0^{(v)}}{85h^6}\right] +$$
$$p_5(x_2)\left[\frac{863y_0-1620y_1+1140y_2-440y_3+45y_4+12y_5+420hy_0'-11h^5y_0^{(v)}}{85h^5}\right] +$$
$$p_4(x_2)\left[\frac{y_0-4y_1+6y_2-4y_3+y_4}{h^4}\right] + p_3(x_2)\left[\frac{-y_0+2y_1-2y_3+y_4}{2h^3}\right] + p_2(x_2)\left[\frac{y_1-2y_2+y_3}{h^2}\right] +$$
$$p_1(x_2)\left[\frac{y_3-y_1}{2h}\right] + p_0(x_2)y(x_2) + r(x_2) - \frac{h^2}{1224}[102p_1(x_2)y'''(\xi_2) +$$
$$204p_2(x_2)y^{(iv)}(\eta_2) + 306p_3(x_2)y^{(v)}(\zeta_2) + 204p_4(x_2)y^{(vi)}(\varphi_2) +$$
$$168hp_5(x_2)y^{(viii)}(\rho_2) + 342p_6(x_2)y^{(viii)}(\gamma_2) - 289y^{(ix)}(\omega_2)] \quad ...(4.127)$$

By omitting the error term and multiplying the previous equation by $12240h^7$ we get the finite-difference formula at *i*=2

$-[1395275 - 138960hp_6(x_2) + 124272h^2p_5(x_2) + 12240h^3p_4(x_2) - 6120h^4p_3(x_2)]w_0 + [2056320 - 306720hp_6(x_2) + 233280h^2p_5(x_2) + 48960h^3p_4(x_2) - 12240h^4p_3(x_2) - 12240h^5p_2(x_2) + 6120h^6p_1(x_2)]w_1 - [963900 - 311040hp_6(x_2) + 164160h^2p_5(x_2) - 73440h^3p_4(x_2) - 24480h^5p_2(x_2) + 12240h^7p_0(x_2)]w_2 + [380800 - 210240hp_6(x_2) + 63360h^2p_5(x_2) + 48960h^3p_4(x_2) + 12240h^4p_3(x_2) - 12240h^5p_2(x_2) - 6120h^6p_1(x_2)]w_3 - [80325 - 79920hp_6(x_2) + 6480h^2p_5(x_2) + 12240h^3p_4(x_2) + 6120h^4p_3(x_2)]w_4 - [12960hp_6(x_2) + 1728h^2p_5(x_2)]w_5 + 2380w_6 - [963900 - 60480hp_6(x_2) + 60480h^2p_5(x_2)]hw_0' - 214200h^2w_0'' + [864hp_6(x_2) + 1584h^2p_5(x_2)]h^5w_0^{(v)} = 12240h^7r(x_2)$ …(4.128)

Now we will derive the finite-difference formula at i=3, we will use the equations (4.25), (4.26), (4.27) and (4.28) for first up to fourth-derivative and we will use

$$y_3^{(v)} = \frac{-y_0+4y_1-5y_2+5y_4-4y_5+y_6}{2h^5} - \frac{h^2}{3}y^{(vii)}(\rho_3) \quad \ldots(4.129)$$

$$y_3^{(vi)} = \frac{y_0-6y_1+15y_2-20y_3+15y_4-6y_5+y_6}{h^6} - \frac{h^2}{6}y^{(viii)}(\gamma_3) \quad \ldots(4.130)$$

$$y_3^{(vii)} = \frac{1008y_0-2400y_1+2865y_2-2440y_3+1320y_4-408y_5+55y_6+420hy_0'-6h^5y_0^{(v)}}{35h^7} + \frac{19h^2}{840}y^{(ix)}(\omega_3) \quad \ldots(4.131)$$

By substituting equations (4.25), (4.26), (4.27), (4.28), (4.129), (4.130) and (4.131) in Eq.(4.112) we get

$\frac{1008y_0-2400y_1+2865y_2-2440y_3+1320y_4-408y_5+55y_6+420hy_0'-6h^5y_0^{(v)}}{35h^7} = p_6(x_3)\left[\frac{y_0-6y_1+15y_2-20y_3+15y_4-6y_5+y_6}{h^6}\right] + p_5(x_3)\left[\frac{-y_0+4y_1-5y_2+5y_4-4y_5+y_6}{2h^5}\right] + p_4(x_3)\left[\frac{y_1-4y_2+6y_3-4y_4+y_5}{h^4}\right] + p_3(x_3)\left[\frac{-y_1+2y_2-2y_4+y_5}{2h^3}\right] + p_2(x_3)\left[\frac{y_2-2y_3+y_4}{h^2}\right] + p_1(x_3)\left[\frac{y_4-y_2}{2h}\right] + p_0(x_3)y_3 + r(x_3) - \frac{h^2}{840}[70p_1(x_3)y'''(\xi_3) +$

$140p_2(x_3)y^{(iv)}(\eta_3) + 210p_3(x_3)y^{(v)}(\zeta_3) + 140p_4(x_3)y^{(vi)}(\varphi_3) + 280hp_5(x_3)y^{(vii)}(\rho_3) + 210p_6(x_3)y^{(viii)}(\gamma_3) + 19y^{(ix)}(\omega_3)]$ ...(4.132)

By omitting the error term and multiplying the previous equation by $70h^7$ we get the finite-difference formula at i=3

$[2016 - 70hp_6(x_3) + 35h^2p_5(x_3)]w_0 - [4800 - 420hp_6(x_3) + 140h^2p_5(x_3) + 70h^3p_4(x_3) - 35h^4p_3(x_2)]w_1 + [5730 - 1050hp_6(x_3) + 175h^2p_5(x_3) - 280h^3p_4(x_3) + 70h^4p_3(x_3) - 70h^5p_2(x_3) + 35h^6p_1(x_3)]w_2 - [4880 - 1400hp_6(x_3) + 420h^3p_4(x_3) - 140h^5p_2(x_3) + 70h^7p_0(x_3)]w_3 + [2640 - 1050hp_6(x_3) - 175h^2p_5(x_3) + 280h^3p_4(x_3) + 70h^4p_3(x_3) - 70h^5p_2(x_3) - 35h^6p_1(x_3)]w_4 - [816 - 420hp_6(x_3) - 140h^2p_5(x_3) + 70h^3p_4(x_3) + 35h^5p_3(x_3)]w_5 + [110 - 70hp_6(x_3) - 35h^2p_5(x_3)]w_6 + 840hw_0' - 12h^5w_0^{(v)} = 70h^7r(x_3)$ …(4.133)

To derive a suitable formula at i=4 we use the following equations which are the centered finite-difference formulas for first up to sixth derivatives at i=4 and a special formula for seventh derivative that maintains second order convergence and satisfies the boundary conditions (4.113) using Algorithm(1.2) many times with several permutations of using boundary conditions to get best local truncation error for the formula.

$$y_4' = \frac{y_5-y_3}{2h} - \frac{h^2}{12}y'''(\xi_4) \qquad \text{…(4.134)}$$

$$y_4'' = \frac{y_3-2y_4+y_5}{h^2} - \frac{h^2}{6}y^{(iv)}(\eta_4) \qquad \text{…(4.135)}$$

$$y_4''' = \frac{-y_2+2y_3-2y_5+y_6}{2h^3} - \frac{h^2}{4}y^{(v)}(\zeta_4) \qquad \text{…(4.136)}$$

$$y_4^{(iv)} = \frac{y_2-4y_3+6y_4-4y_5+y_6}{h^4} - \frac{h^2}{6}y^{(vi)}(\varphi_4) \qquad \text{…(4.137)}$$

$$y_4^{(v)} = \frac{-y_1+4y_2-5y_3+5y_5-4y_6+y_7}{2h^5} - \frac{h^2}{3}y^{(vii)}(\rho_4) \qquad \text{…(4.138)}$$

$$y_4^{(vi)} = \frac{y_1-6y_2+15y_3-20y_4+15y_5-6y_6+y_7}{h^6} - \frac{h^2}{4}y^{(viii)}(\gamma_4) \qquad \ldots(4.139)$$

$$y_4^{(vii)} =$$

$$\frac{-1194y_0+3675y_1-6615y_2+8575y_3-7350y_4+3969y_5-1225y_6+165y_7-420hy_0'}{105h^7} + \frac{h^2}{36}y^{(ix)}(\omega_4)$$

$$\ldots(4.140)$$

By substituting equations (4.134),(4.135),(4.136),(4.137),(4.138),(4.139) and (4.140) in Eq.(4.112) we get

$$\frac{-1194y_0+3675y_1-6615y_2+8575y_3-7350y_4+3969y_5-1225y_6+165y_7-420hy_0'}{105h^7} =$$
$$p_6(x_4)\left[\frac{y_1-6y_2+15y_3-20y_4+15y_5-6y_6+y_7}{h^6}\right] + p_5(x_4)\left[\frac{-y_1+4y_2-5y_3+5y_5-4y_6+y_7}{2h^5}\right] +$$
$$p_4(x_4)\left[\frac{y_2-4y_3+6y_4-4y_5+y_6}{h^4}\right] + p_3(x_4)\left[\frac{-y_2+2y_3-2y_5+y_6}{2h^3}\right] + p_2(x_4)\left[\frac{y_3-2y_4+y_5}{h^2}\right] +$$
$$p_1(x_3)\left[\frac{y_5-y_3}{2h}\right] + p_0(x_4)y_4 + r(x_4) - \frac{h^2}{36}[3p_1(x_4)y'''(\xi_4) + 6p_2(x_4)y^{(iv)}(\eta_4) +$$
$$9p_3(x_4)y^{(v)}(\zeta_4) + 6p_4(x_4)y^{(vi)}(\varphi_4) + 12hp_5(x_4)y^{(vii)}(\rho_4) +$$
$$9p_6(x_4)y^{(viii)}(\gamma_4) + y^{(ix)}(\omega_4)] \qquad \ldots(4.141)$$

By omitting the error term and multiplying the previous equation by $210h^7$ we get the finite-difference formula at *i*=4

$$-2388w_0 + [7350 - 210hp_6(x_4) + 105h^2p_5(x_4)]w_1 - [13230 - 1260hp_6(x_4) +$$
$$420h^2p_5(x_4) + 210h^3p_4(x_4) - 105h^4p_3(x_4)]w_2 + [17150 - 3150hp_6(x_4) +$$
$$525h^2p_5(x_4) - 840h^3p_4(x_4) + 210h^4p_3(x_4) - 210h^5p_2(x_4) + 105h^6p_1(x_4)]w_3 -$$
$$[14700 - 4200hp_6(x_4) + 1260h^3p_4(x_4) - 420h^5p_2(x_4) + 210h^7p_0(x_4)]w_4 +$$
$$[7938 - 3150hp_6(x_4) - 525h^2p_5(x_4) + 840h^3p_4(x_4) + 210h^4p_3(x_4) -$$
$$210h^5p_2(x_4) - 105h^6p_1(x_4)]w_5 - [2450 - 1260hp_6(x_4) - 420h^2p_5(x_4) +$$
$$210h^3p_4(x_4) + 105h^5p_3(x_4)]w_6 + [330 - 210hp_6(x_4) - 105h^2p_5(x_4)]w_7 -$$
$$840hw_0' = 210h^7r(x_4) \qquad \ldots(4.142)$$

To find the suitable finite-difference formula at i=N-1, we use the equations (4.57),(4.58),(4.59)and (4.60) that are the centered finite difference approximations at i=N-1and derive a new formula for fifth, sixth and seventh-order derivatives at i=N-1 with one more point for seventh-order

$$y_{N-1}^{(v)}=\frac{-12y_{N-4}-45y_{N-3}+440y_{N-2}-1140y_{N-1}+1620y_N-863y_{N+1}+420hy'_{N+1}-11h^5y_{N+1}^{(v)}}{85h^5}+\frac{7h^3}{51}y^{(viii)}(\rho_{N-1}) \quad \ldots(4.143)$$

$$y_{N-1}^{(vi)}=\frac{90y_{N-4}-555y_{N-3}+1460y_{N-2}-2160y_{N-1}+2130y_N-965y_{N+1}+420hy'_{N+1}+6h^5y_{N+1}^{(v)}}{85h^6}-\frac{19h^2}{68}y^{(viii)}(\gamma_{N-1}) \quad \ldots(4.144)$$

$$y_{N-1}^{(vii)}=\frac{25w_{N-6}-245w_{N-5}+1029y_{N-4}-2450y_{N-3}+3675y_{N-2}-3675y_{N-1}+2695y_N-1054y_{N+1}+420hy'_{N+1}}{35h^7}+\frac{h^2}{4}y^{(ix)}(\omega_{N-1}) \quad \ldots(4.145)$$

By substituting equations (4.57),(4.58),(4.59),(4.60),(4.143),(4.144) and (4.145) in Eq.(4.112) we get

$$\begin{aligned}&\frac{25w_{N-6}-245w_{N-5}+1029y_{N-4}-2450y_{N-3}+3675y_{N-2}-3675y_{N-1}+2695y_N-1054y_{N+1}+420hy'_{N+1}}{35h^7}=\\&p_6(x_{N-1})\left[\frac{90y_{N-4}-555y_{N-3}+1460y_{N-2}-2160y_{N-1}+2130y_N-965y_{N+1}+420hy'_{N+1}+6h^5y_{N+1}^{(v)}}{85h^6}\right]+\\&p_5(x_{N-1})\left[\frac{-12y_{N-4}-45y_{N-3}+440y_{N-2}-1140y_{N-1}+1620y_N-863y_{N+1}+420hy'_{N+1}-11h^5y_{N+1}^{(v)}}{85h^5}\right]+\\&p_4(x_{N-1})\left[\frac{y_{N-3}-4y_{N-2}+6y_{N-1}-4y_N+y_{N+1}}{h^4}\right]+p_3(x_{N-1})\left[\frac{-y_{N-3}+2y_{N-2}-2y_N+y_{N+1}}{2h^3}\right]+\\&p_2(x_{N-1})\left[\frac{y_{N-2}-2y_{N-1}+y_N}{h^2}\right]+p_1(x_{N-1})\left[\frac{y_{N-2}-y_N}{2h}\right]+p_0(x_{N-1})y_{N-1}+r(x_N)-\\&\frac{h^2}{204}[17p_1(x_{N-1})y'''(\xi_{N-1})+34p_2(x_{N-1})y^{(iv)}(\eta_{N-1})+51p_3(x_{N-1})y^{(v)}(\zeta_{N-1})+\\&34p_4(x_{N-1})y^{(vi)}(\varphi_{N-1})-28hp_5(x_{N-1})y^{(viii)}(\rho_{N-1})+57p_6(x_{N-1})y^{(viii)}(\gamma_{N-1})+\\&51y^{(ix)}(\omega_{N-1})\end{aligned} \quad \ldots(4.146)$$

By omitting the error term and multiplying the previous equation by $1190h^7$ we get the finite-difference formula at i=N-1

$850w_{N-6} - 8330w_{N-5} + [34986 - 1260hp_6(x_{N-1}) + 168h^2p_5(x_{N-1})]w_{N-4} - [83300 - 7770hp_6(x_{N-1}) - 630h^2p_5(x_{N-1}) + 1190h^3p_4(x_{N-1}) - 595h^4p_3(x_{N-1})]w_{N-3} + [124950 - 20440hp_6(x_{N-1}) - 6160h^2p_5(x_{N-1}) + 2380h^3p_4(x_{N-1}) - 1190h^4p_3(x_{N-1}) - 1190h^5p_2(x_{N-1}) + 595h^6p_1(x_{N-1})]w_{N-2} - [124950-30240hp_6(x_{N-1}) - 15960h^2p_5(x_{N-1}) + 7140h^3p_4(x_{N-1}) - 2380h^5p_2(x_{N-1}) + 1190h^7p_0(x_{N-1})]w_{N-1} + [91630 - 29820hp_6(x_{N-1}) - 22680h^2p_5(x_{N-1}) + 4760h^3p_4(x_{N-1}) + 1190h^4p_3(x_{N-1}) - 1190h^5p_2(x_{N-1}) - 595h^6p_1(x_{N-1})]w_N - [35836 - 13510hp_6(x_{N-1}) - 12082h^2p_5(x_{N-1}) + 1190h^3p_4(x_{N-1}) + 595h^4p_3(x_{N-1})]w_{N+1} + [14280 - 5880hp_6(x_{N-1}) - 5880h^2p_5(x_{N-1})]hw'_{N+1} - [84hp_6(x_{N-1}) - 154h^2p_5(x_{N-1})]h^5w_{N+1}^{(v)} = 1190h^7r(x_{N-1})$ …(4.147)

Finally we derive the suitable finite –difference formula at i=N using equation (4.32) & (4.33) for first and second derivative and the following equations for third up to seventh derivative using the boundary conditions (4.113).

$$y_N''' = \frac{-20y_{N-2}-180y_N-160y_{N+1}+120hw'_{N+1}-9h^5y_{N+1}^{(v)}}{60h^3} - \frac{h^3}{6}y^{(vi)}(\zeta_N) \quad \ldots(4.148)$$

$$y_N^{(iv)} = \frac{20y_{N-2}-90y_{N-1}+180y_N-110y_{N+1}+60hw'_{N+1}+3h^5y_{N+1}^{(v)}}{15h^4} - \frac{h^2}{3}y^{(viii)}(\varphi_N) \quad \ldots(4.149)$$

$$y_N^{(v)} = \frac{-15y_{N-3}+80y_{N-2}-180y_{N-1}+240y_N-125y_{N+1}+60hw'_{N+1}+8h^5y_{N+1}^{(v)}}{20h^5} - \frac{3h^2}{7}y^{(vii)}(\rho_{N-1}) \quad \ldots(4.150)$$

$$y_N^{(vi)} = \frac{12y_{N-4}+45y_{N-3}-440y_{N-2}+1140y_{N-1}-1620y_N+863y_{N+1}-420hw'_{N+1}+96h^5y_{N+1}^{(v)}}{170h^6} - \frac{4h^2}{17}y^{(viii)}(\gamma_N) \quad \ldots(4.151)$$

$$y_N^{(vii)} = \frac{185w_{N-5}-1800y_{N-4}+7380y_{N-3}-16760y_{N-2}+23535y_{N-1}-23040y_N+10500y_{N+1}-4620hw'_{N+1}+234h^5y_{N+1}^{(v)}}{315h^7} + \frac{369h^2}{1861}y^{(ix)}(\omega_N) \quad \ldots(4.152)$$

Substituting equations (4.32), (4.33), (4.148), (4.149), (4.150), (4.151) and (4.152) in equation (4.112) we get the equation

$$\frac{185w_{N-5}-1800y_{N-4}+7380y_{N-3}-16760y_{N-2}+23535y_{N-1}-23040y_N+10500y_{N+1}-4620hw'_{N+1}+234h^5y_{N+1}^{(v)}}{315h^7} =$$

$$p_6(x_N)\left[\frac{12y_{N-4}+45y_{N-3}-440y_{N-2}+1140y_{N-1}-1620y_N+863y_{N+1}-420hw'_{N+1}+96h^5y_{N+1}^{(v)}}{170h^6}\right]+$$

$$p_5(x_N)\left[\frac{-15y_{N-3}+80y_{N-2}-180y_{N-1}+240y_N-125y_{N+1}+60hw'_{N+1}+8h^5y_{N+1}^{(v)}}{20h^5}\right]+$$

$$p_4(x_N)\left[\frac{20y_{N-2}-90y_{N-1}+180y_N-110y_{N+1}+60hw'_{N+1}+3h^5y_{N+1}^{(v)}}{15h^4}\right]+$$

$$p_3(x_N)\left[\frac{-20y_{N-2}-180y_N-160y_{N+1}+120hw'_{N+1}-9h^5y_{N+1}^{(v)}}{60h^3}\right]+p_2(x_N)\left[\frac{y_{N-1}-2y_N+y_{N+1}}{h^2}\right]+$$

$$p_1(x_N)\left[\frac{y_{N-1}-y_{N+1}}{2h}\right]+p_0(x_N)y_N+r(x_N)-\frac{h^2}{2657508}[221459p_1(x_N)y'''(\xi_N)+$$

$$442918p_2(x_N)y^{(iv)}(\eta_N)+442918hp_3(x_N)y^{(vi)}(\zeta_N)+885836p_4(x_N)y^{(vi)}(\varphi_N)+$$

$$1138932p_5(x_N)y^{(vii)}(\rho_N)+625296p_6(x_N)y^{(viii)}(\gamma_N)+526932y^{(ix)}(\omega_N)]$$

…(4.153)

By omitting the error term and multiplying the previous equation by $21420h^7$ we get the finite-difference formula at i=N

$$12580w_{N-5}-[122400+1512hp_6(x_N)]w_{N-4}+[501840-5670hp_6(x_N)+$$
$$16065h^2p_5(x_N)]w_{N-3}-[1139680-55440hp_6(x_N)+85680h^2p_5(x_N)+$$
$$28560h^3p_4(x_N)-7140h^4p_3(x_N)]w_{N-2}+[1600380-143640hp_6(x_N)+$$
$$192780h^2p_5(x_N)+128520h^3p_4(x_N)-21420h^5p_2(x_N)+10710h^6p_1(x_N)]w_{N-1}-$$
$$[1566720-204120hp_6(x_N)+257040h^2p_5(x_N)+257040h^3p_4(x_N)-$$
$$64260h^4p_3(x_N)-42840h^5p_2(x_N)+21420h^7p_0(x_N)]w_N+[714000-$$
$$108738hp_6(x_N)+133875h^2p_5(x_N)+157080h^3p_4(x_N)+57120h^4p_3(x_N)-$$
$$21420h^5p_2(x_N)-10710h^6p_1(x_N)]w_{N+1}-[314160-52920hp_6(x_N)+$$
$$64260h^2p_5(x_N)+85680h^3p_4(x_N)+42840h^4p_3(x_N)]hw'_{N+1}+[15912-$$
$$12096hp_6(x_N)-8568h^2p_5(x_N)+4284h^3p_4(x_N)+3213h^4p_3(x_N)]h^5w_{N+1}^{(v)}=$$
$$21420h^7r(x_N)$$

…(4.154)

Using equations (4.117),(4.123),(4.128),(4.133),(4.142),(5.147),and (4.154) together gives a system with $N\times N$ nearly tenth-diagonal matrix problem

$$Aw = \mathrm{C} \qquad \ldots(4.155)$$

Where $A=$

$$\begin{bmatrix}
(-552960-G_{11}) & -(564840+G_{12}) & (402240-G_{13}) & -(177120+G_{14}) & 43200 & -4440 & & 0 \\
(205632-G_{21}) & -963900-G_{22} & (380800-G_{23}) & -80325-G_{24} & (-G_{25}) & 2380 & & \\
-(4800+35F_{23)} & 5730-35F_{33} & -(4880+35F_{43}) & 2640-35F_{53} & -(816+35F_{63}) & 110-35F_{13} & & \\
7350-105F_{14} & -(13230+105F_{24}) & 17150-105F_{34} & -(14700+105F_{44}) & 7938-105F_{54} & -(2450+F_{64}) & 330-35F_{74} & \\
-10 & 42-F_{15} & -(98+F_{25}) & 146-F_{35} & -(126+F_{45}) & 70-F_{55} & -(22+F_{65}) & 3-F_{75} \\
 & & & & & & & \\
 & & 2-F_{1N-2} & -(12+F_{2N-2}) & (30-F_{3N-2}) & -(40+F_{4,N-2}) & (30+F_{5,N-2}) & -(12+F_{6,N-2}) \\
 & 850 & -8330 & (34986-G_{N11}) & -(83300+G_{N12}) & (124950-G_{N13}) & -(124950+G_{N14}) & 91630-G_{N15} \\
0 & & 185 & -(122400+G_{N1}) & (501840-G_{N2}) & -(1139680+G_{N3}) & (1600380-G_{N4}) & -(1566720+G_{N5})
\end{bmatrix}_{N\times N}$$

$$w = [w_1 \quad w_2 \quad \ldots \quad w_{N-1} \quad w_N]_{1\times N}{}^{T} \ ,$$

$$c =$$

$$\begin{bmatrix}
(252000+G_{10})\alpha_0+(110830+G_{101})h\alpha_1+G_{102}h^2\alpha_2+(-5616+G_{105})h^5\alpha_5+7560h^7r(x_1) \\
(1395275+G_{20})\alpha_0+(963900+G_{201})h\alpha_1+214200h^2\alpha_2+(G_{205})h^5\alpha_5+12240h^7r(x_2) \\
(-2016+35F_{13})\alpha_0-840h\alpha_1-12h^5\alpha_5+70h^7r(x_3) \\
2388\alpha_0+840h^5\alpha_5+210h^7r(x_4) \\
-\alpha_0+2h^7r(x_5) \\
2h^7r(x_6) \\
\vdots \\
-(3-F_{7\,N-2})\beta_0+2h^7r(x_{N-2}) \\
(35836+G_{N10})\beta_0+(-14280+G_{N101})h\beta_1+G_{N105}h^5\beta_5+1190h^7r(x_{N-1}) \\
(-71400+G_{N0})\beta_0++(314160+G_{N01})h\beta_1-(15912+G_{N05})h^5\beta_5+21420h^7r(x_N)
\end{bmatrix}_{N\times 1}$$

$$F_{1i} = 2hp_6(x_i) - h^2p_5(x_i)\ , \quad F_{2i} = -12hp_6(x_i) - 4h^2p_5(x_i) - 2h^3p_4(x_i) - h^4p_3(x_i)$$

$$F_{3i} = 30hp_6(x_i) - 5h^2p_5(x_i) - 8h^3p_4(x_i) + 2h^4p_3(x_i) + 2h^5p_2(x_i) - h^6p_1(x_i)$$

$$F_{4i} = -40hp_6(x_i) + 12h^3p_4(x_i) - 4h^5p_2(x_i) + 2h^7p_0(x_i)\ ,$$

$$F_{5i} = 30hp_6(x_i) + 5h^2p_5(x_i) - 8h^3p_4(x_i) - 2h^4p_3(x_i) + 2h^5p_2(x_i) + h^6p_1(x_i)\ ,$$

$$F_{6i} = -12hp_6(x_i) - 4h^2p_5(x_i) + 2hp_4(x_i) + h^2p_3(x_i)\ , F_{7i} = 2hp_6(x_i) + h^2p_5(x_i)$$

$G_0 = [133875hp_6(x_1) - 472500h^2p_5(x_1) - 63560h^3p_4(x_1) + 73710h^4p_3(x_1) + 7560h^5p_2(x_1) - 3780h^6p_1(x_1)]$

$G_1 = -[207360hp_6(x_1) - 609120h^2p_5(x_1) + 60480h^3p_4(x_1) + 90720h^4p_3(x_1) + 15120h^5p_2(x_1) - 7560h^7p_0(x_1)]$

$G_2 = [106920hp_6(x_1) - 165240h^2p_5(x_1) - 7560h^3p_4(x_1) + 17010h^4p_3(x_1) + 7560h^5p_2(x_1) + 3780h^6p_1(x_1)]$

$G_3 = -[40320hp_6(x_1) - 30240h^2p_5(x_1) - 4480h^3p_4(x_1)]$

$G_4 = [6885hp_6(x_1) - 1620h^2p_5(x_1)]w_4 + 43200w_5 - 4440w_6$

$G_{01} = [86940hp_6(x_1) - 362880h^2p_5(x_1) - 62160h^3p_4(x_1) + 56700 - 30240h^4p_3(x_1)]$

$G_{02} = [16200hp_6(x_1) - 97200h^2p_5(x_1) + 25200h^3p_4(x_1) + 11340h^4p_3(x_1)]$

$G_{05} = -[3888hp_6(x_1) + 864h^2p_5(x_1) - 1008h^3p_4(x_1)]$

$G_{20} = -[-138960hp_6(x_2) + 124272h^2p_5(x_2) + 12240h^3p_4(x_2) - 6120h^4p_3(x_2)]$

$G_{21} = [-306720hp_6(x_2) + 233280h^2p_5(x_2) + 48960h^3p_4(x_2) - 12240h^4p_3(x_2) - 12240h^5p_2(x_2) + 6120h^6p_1(x_2)]$

$G_{22} = -[-311040hp_6(x_2) + 164160h^2p_5(x_2) - 73440h^3p_4(x_2) - 24480h^5p_2(x_2) + 12240h^7p_0(x_2)]$

$G_{23} = -[-210240hp_6(x_2) + 63360h^2p_5(x_2) + 48960h^3p_4(x_2) + 12240h^4p_3(x_2) - 12240h^5p_2(x_2) - 6120h^6p_1(x_2)]$

$G_{24} = [-79920hp_6(x_2) + 6480h^2p_5(x_2) + 12240h^3p_4(x_2) + 6120h^4p_3(x_2)]$

$G_{20} = [12960hp_6(x_2) + 1728h^2p_5(x_2)]$ $G_{201} = [963900 - 60480hp_6(x_2) + 60480h^2p_5(x_2)]$

$G_{205} = -[864hp_6(x_2) + 1584h^2p_5(x_2)]$

$G_{N11} = -\left[-1260hp_6(x_{N-1}) + 168h^2p_5(x_{N-1})\right]$

$G_{N12} = [-7770hp_6(x_{N-1}) - 630h^2p_5(x_{N-1}) + 1190h^3p_4(x_{N-1}) - 595h^4p_3(x_{N-1})]$

$G_{N13} = [-20440hp_6(x_{N-1}) - 6160h^2p_5(x_{N-1}) + 2380h^3p_4(x_{N-1}) - 1190h^4p_3(x_{N-1}) - 1190h^5p_2(x_{N-1}) + 595h^6p_1(x_{N-1})]$

$G_{N14} = [-30240hp_6(x_{N-1}) - 15960h^2p_5(x_{N-1}) + 7140h^3p_4(x_{N-1}) - 2380h^5p_2(x_{N-1}) + 1190h^7p_0(x_{N-1})]$

$G_{N15} = -[-29820hp_6(x_{N-1}) - 22680h^2p_5(x_{N-1}) + 4760h^3p_4(x_{N-1}) + 1190h^4p_3(x_{N-1}) - 1190h^5p_2(x_{N-1}) - 595h^6p_1(x_{N-1})]$

$G_{N10} = [-13510hp_6(x_{N-1}) - 12082h^2p_5(x_{N-1}) + 1190h^3p_4(x_{N-1}) + 595h^4p_3(x_{N-1})]$,
$G_{N101} = -\left[-5880hp_6(x_{N-1}) - 5880h^2p_5(x_{N-1})\right]$

$G_{N105} = [84hp_6(x_{N-1}) - 154h^2p_5(x_{N-1})]$

$G_{N1} = [1512hp_6(x_N)], G_{N2} = -[-5670hp_6(x_N) + 16065h^2p_5(x_N)]$,

$G_{N3} = [-55440hp_6(x_N) + 85680h^2p_5(x_N) + 28560h^3p_4(x_N) - 7140h^4p_3(x_N)]$,

$G_{N4} =$

$-[-143640hp_6(x_N) + 192780h^2p_5(x_N) + 128520h^3p_4(x_N) - 21420h^5p_2(x_N) + 10710h^6p_1(x_N)]$

$G_{N5} =$

$[-204120hp_6(x_N) + 257040h^2p_5(x_N) + 257040h^3p_4(x_N) - 64260h^4p_3(x_N) - 42840h^5p_2(x_N) + 21420h^7p_0(x_N)]$

$G_{N0} =$

$-[-108738hp_6(x_N) + 133875h^2p_5(x_N) + 157080h^3p_4(x_N) + 57120h^4p_3(x_N) - 21420h^5p_2(x_N) - 10710h^6p_1(x_N)]$

$G_{N01} = [-52920hp_6(x_N) + 64260h^2p_5(x_N) + 85680h^3p_4(x_N) + 42840h^4p_3(x_N)]$

$G_{N05} = -[-12096hp_6(x_N) - 8568h^2p_5(x_N) + 4284h^3p_4(x_N) + 3213h^4p_3(x_N)]$

Now we use the LU-decomposition method with MATLAB for solving the linear system (4.111) at several values of step-size (h) and then we use Extrapolation method to get more accuracy for the method without more calculations.

Algorithm (4.5): Linear Finite-Difference Method (4.5) For Solving Seventh-Order BVPS Case (I): See full MATLAB program (p.262-276)

To approximate the solution of the boundary value problem

$y^{(vii)}(x) =$

$p_6(x)y^{(vi)} + p_5(x)y^{(v)} + p_4(x)y^{(iv)} + p_3(x)y'''(x) + p_2(x)y'' + p_1(x)y' + p_0(x)y + r(x)$

$for\ x, \quad a \le x \le b$ With seven boundary conditions

$$y(a) = \alpha_0 \ , y'(a) = \alpha_1 \, , y''(a) = \alpha_2 , y^{(v)}(a) = \alpha_5 \, ,$$

$$y(b) = \beta_0 \ , y'(b) = \beta_1 \quad , y^{(v)}(b) = \beta_5$$

Step(1):Input endpoints a,b; boundary conditions $\alpha_0, \alpha_1, \alpha_2, \alpha_5, \beta_0, \beta_1, \beta_5$

Step(2): For k=2,3,4 (To determine h=0.1,0.05,0.025,0,0125)

- set $N_k=(10*2^{(k-1)})-1$;
- $h=(b-a)/(N_k+1)$;
- do steps(2-9)

Step(3): For $i=1,2,3, \ldots, N_k$ set $x(i)=a+ih$

Find $p_0(x_i), p_1(x_i), p_2(x_i), p_3(x_i) \, , p_4(x_i), p_5(x_i), p_6(x_i) and \ r(x_i)$.

(The values of associated functions on Eq.(4.112)

Step(4): For $i=1,2,3,\ldots,N_k$ set

d0=the vector of entries for main diagonal of matrix A

dU1=the vector of entries for the first upper diagonal for A

dU2= the vector of entries for the second upper diagonal for A

dU3= the vector of entries for the third upper diagonal for A

dU4= the vector of entries for the fourth upper diagonal for A

dU5= the vector of entries for the fifth upper diagonal for A

dL1= the vector of entries for the first lower diagonal for A

dL2= the vector of entries for the second lower diagonal for A

dL3= the vector of entries for the third lower diagonal for A

dL4= the vector of entries for the fourth lower diagonal for A

dL5= the vector of entries for the fifth lower diagonal for A

dL6= the vector of entries for the sixth lower diagonal for A

C=the constant vector on (c) Eq.(4.155)

Step(6) Factorize A into LU, where L is lower sixth-diagonal matrix with unit main diagonal and U is upper sixth-diagonal.

Step(7) For $i=1,2,3,\ldots,N_k$ Solve LY=d to find Y using forward method

Step(8) For $i=N_k-1\ ,..2,1$ Solve Uw=Y to find the approximation solution w using backward method

Step(9) Set N1=9 For $i=1,2,3,\ldots N_k$, for j=1,2,4,…,N_k set $W1_{i,k}=w_j$;

Step(10) For $i=1,2,\ldots,N1$ set h=(b-a)/N1 ; x(i)=a+ih

Step(11) Extrapolate the solution at h= .1,0.05,0.025,0,0125

Ext2(h=0.1)=(4*W1(h=0.05)-W1(h=0.1))/3;

Ext2(h=0.05)= (4*W1(h=0.025)-W1(h=0.05))/3;

Ext2(h=0.025)= (4*W1(h=0.0125)-W1(h=0.025))/3;

Ext3(h=0.1)=(16*Ext2(h=0.05)-Ext2(h=0.1))/15;

Ext3(h=0.05)=(16*Ext2(h=0.025)-Ext2(h=0.05)/15;

Ext4(h=0.1)=(64*Ext3(h=0.05)-Ext3(h=0.1))/63;

Step(12) Output [x' ,W1',Ext4] (The approximation solutions and Extrapolation)

Example 5: Consider the linear seventh-order boundary value problem

$y^{(vi)}(x) = x^2y^{(vi)} - (2x^2 - 1)y^{(v)} - (x^3 + x^2 + 1)y^{(iv)} + 5x^3y''' + 9xy'' - (1 - x)y' - (3 + 2x)y + 4e^{2x}(4x^4 - 3x^3 + 13x^2 - 29x - 71)$ on $0 \leq x \leq 1$

With boundary conditions:

$y(0) = 1, \quad y'(0) = 1, \quad y''(0) = 0\,, \; y^{(v)}(0) = -48$

$y(1) = 0, y'(1) = -e^2\,, y^{(v)}(1) = -80e^2$

So $p_6(x) = x^2, p_5(x) = -(2x^2 - 1)\,, p_4(x) = -(x^3 + x^2 + 1),$

$p_3(x) = 5x^3, p_2(x) = 9x\,, p_1(x) = -(1 - x), p_0(x) = -(3 + 2x)$ And

$r(x) = 4e^{2x}(4x^4 - 3x^3 + 13x^2 - 29x - 71),$ on the interval [*0,1*]

Now by applying algorithm (4.5) to the example we get the following results that approximated solutions at several values of h as listed in Table (4.9)

Table 4.9

X_i	Exact(Y)	w(h=0.05)	w(h=0.025)	w(h=0.0125)	w(h=0.0625)	Ext3(h=0.05)	Ext4(h=0.05)
0.1	1.09926248	1.09719711	1.10098335	1.11011953	1.12901734	1.11389288	1.13715688
0.2	1.19345976	1.19706218	1.22389714	1.28171988	1.39897242	1.30553759	1.44944261
0.3	1.27548316	1.29835857	1.36887006	1.51585274	1.81202645	1.57634517	1.93948536
0.4	1.33532456	1.38589853	1.50804434	1.75870995	2.26225104	1.86183219	2.47892907
0.5	1.35914091	1.43528422	1.59865041	1.93089270	2.59709146	2.06754242	2.88374539
0.6	1.32804677	1.41671207	1.59405539	1.95262027	2.67073972	2.10007337	2.97972211
0.7	1.21655999	1.29774294	1.45277668	1.76496186	2.38966750	1.89332818	2.65844968
0.8	0.99060648	1.04494540	1.14538311	1.34703034	1.75029102	1.42993832	1.92379205
0.9	0.60496475	0.62427555	0.65911995	0.72891955	0.86844117	0.75761618	0.92846876

The maximum errors of approximations are listed in Table(4.10)

Table 4.10

$\|Y-W\|$	**Maximum Error**
$\|Y-w_i(h=0.05)\|$	8.8665×10^{-2}
$\|Y-w_i(h=0.025)\|$	2.6601×10^{-1}
$\|Y-w_i(h=0.0125)\|$	6.2457×10^{-1}
$\|Y-w_i(h=0.0625)\|$	1.3427
$\|Y-Ext3_i(h=0.05)\|$	3.2512×10^{-1}
$\|Y-Ext4_i(h=0.05)\|$	1.6517

This table reveals that the errors increase while the step-size h decrease, so the results for this method are not good as we want for the seventh-order boundary-value problem.

The reasons for these large errors depend on the boundary-conditions and derivatives that are used in the problem.

We use this method using other boundary-conditions ($\alpha_0, \alpha_1, \alpha_3, \alpha_5, \beta_0, \beta_4, and\ \beta_6$) we get very bad results with (5.5676) value of the smallest error is at h=0.05. So the order of derivatives in the boundary-conditions affects strongly the results.

We note in this method that the best boundary-conditions that decrease the more points used in the approximations and increase the convergence for the method are that we use in our research.

CHAPTER FIVE

Eighth-order Non-Linear Boundary-Value Problems and Developed Methods for Solving Some Even-Order Boundary-Value Problems

The Non-linear boundary value problems have many uses and applications in scientific branches, and many researches discussed these problems see [3].

In this chapter we deal with especial case of non-linear boundary value problems which are the even-order problems and especial cases of eighth-order problems. Completing to rare previous studies of these problems order types we modified it approximating solution methods to be suitable for some even-order problems less than eighth-order using new algorithms.

The main idea for solving numerically these kinds of problems is to transform it to system of finite linear equations using set of unknowns and solving it using iterative methods.

5.1 Finite – Difference Method For Solving *2m*-Order Boundary Value Problems:

In this way we will solve *2m*-order boundary value problems numerically by transforming the given differential equation into a system of (m) second –order equations, then we use two-step finite – difference methods for their solution.

Consider special non-linear 2mth-order boundary-value problem:

$$y^{(2m)}(x) = f(x,y) \quad , \quad a < x < b \, , \quad a,b,x \in \mathbb{R} \tag{5.1}$$

$$y^{(2i)}(a) = \alpha_{2i} \quad , y^{(2i)}(b) = \beta_{2i} \quad , i = 0,1,2,\dots,m-1 \tag{5.2}$$

(5.2) are the boundary conditions for (5.1) and $y(x)$,$f(x,y)$ are real and differentiable as required ,and that α_{2i} ,β_{2i} for $i= 0,1,2,\dots,m-1$ are real finite constants.

How are we solving it?

The differential equation (5.1) is replaced by a system of m-second order differential equations:

$$y_{i+1\ (x)} = y_i'' \quad , \quad i = 1,2,3,\dots m-1 \tag{5.3}$$

$$y_m'' = f(x,y_1) \tag{5.4}$$

And the boundary conditions are given by:

$$y_i(a) = \alpha_{2i-2} \quad , \quad y_i(b) = \beta_{2i-2} \quad , i = 1,2,3,\dots,m \tag{5.5}$$

Which mean that if we take sixth-order non-linear boundary value problem:

$y^{(6)}(x) = f(x,y) \quad , \quad a < x < b \quad , a, b, x \in \mathbb{R}$

$y^{(2i)}(a) = \alpha_{2i} \qquad , y^{(2i)}(b) = \beta_{2i} \qquad , i = 0,1,2$

Note that we take the value for even derivatives as boundary conditions value in our especial problem, and by applying (5.3),(5.4) in our problem it will give the system of *m=3* second-order boundary value problems:

$$y_2\ (x) = \ y_1'' \ , \ \ y_1(a) = \alpha_0 \ \ , \ \ y_1(b) = \beta_0 \qquad \ldots .1$$

$$y_3\ (x) = \ y_2'' \ , \ \ y_2(a) = \alpha_2 \ \ , \ \ y_2(b) = \beta_2 \qquad \ldots .2$$

$$y_3'' \ = f(x, y_1) \ , \ \ y_3(a) = \alpha_4 \ \ , \ \ y_3(b) = \beta_4 \qquad \ldots .3$$

Solution On A single Grid G_1:

We select integer *N*>0 and divide the interval [a,b] into *(N+1)* equal subintervals whose endpoints are the mesh points $x_n = a + nh$ for(n= 0,1, 2,3,…,N+1),where $h = (b - a)/(N + 1)$.

The solution $y_i(x)$ will be computed at the mesh point x_n of G_1 and the notation $y_{i,n}$ will denote the solution of an approximating difference method at the grid point x_n .

So $y_{i,0} = y_i(a) = \alpha_{2i-2}$ and $y_{i,N+1} = y_i(b) = \beta_{2i-2}$, i= *1,2,3,…,m.*

Now we introduce a family of numerical methods for solving any second – order problem of the form:

$$y'' = \psi(x, y) \ , x \in [a, b] \quad , y(a) = \alpha_0 \ , y(b) = \beta_0 \qquad (5.6)$$

is given by using centered-difference formula to approximate the derivative as:

$$-y_{i,n-1} + 2y_{i,n} - y_{i,n+1} + h^2\left[\theta y''_{i,n-1} + (1 - 2\theta\ )y''_{i,n} + \theta y''_{i,n+1}\right] = 0 \tag{5.7}$$

Where θ is a parameter with $0 \leq \theta \leq 1$. This determines the order of convergence for the method (5.7) which is second – order convergent for $\theta \neq 0$ and fourth- order convergent for $\theta = \frac{1}{12}$.

Now we use the equation (5.7) and the boundary values in equations (5.3) and omitting error terms to obtain the following system:

For $i=1$

$$-w_{1,0} + 2w_{1,1} - w_{1,2} + h^2\left[\theta w''_{1,0} + (1 - 2\theta)w''_{1,1} + \theta w''_{1,2}\right] = 0$$

$$-w_{1,1} + 2w_{1,2} - w_{1,3} + h^2\left[\theta w''_{1,1} + (1 - 2\theta)w''_{1,2} + \theta w''_{1,3}\right] = 0$$

$$\ddots \qquad \ddots \qquad \ddots \qquad \ddots$$

$$-w_{1,N-1} + 2w_{1,N} - w_{1,N+1} + h^2\left[\theta w''_{1,N-1} + (1 - 2\theta)w''_{1,N} + \theta w''_{1,N+1}\right] = 0$$

Which give the system:

$$\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}\begin{bmatrix} w_{1,1} \\ w_{1,2} \\ \vdots \\ w_{1,N-1} \\ w_{1,N} \end{bmatrix} + \begin{bmatrix} 1-2\theta & \theta & & & \\ \theta & 1-2\theta & \theta & & \\ & \ddots & \ddots & \ddots & \\ & & \theta & 1-2\theta & \theta \\ & & & \theta & 1-2\theta \end{bmatrix}\begin{bmatrix} w_{2,1} \\ w_{2,2} \\ \vdots \\ w_{2,N-1} \\ w_{2,N} \end{bmatrix} = \begin{bmatrix} \alpha_0 - \theta h^2\alpha_2 \\ 0 \\ \vdots \\ 0 \\ \beta_0 - \theta h^2\beta_0 \end{bmatrix}$$

Noting that $w''_i = w_{i+1}$ for i<m, so $w''_1(x_n) = w_2(x_n)$, now this system can be summarized in this formula:

$$J_1W_1 + h^2M_1W_2 = b_1$$

And so on for $i<m$, to a general formula:

$$J_1 W_i + h^2 M_1 W_{i+1} = b_i \qquad \ldots(5.8)$$

The vector $W_i = \left[w_{i,1}, w_{i,2}, \dots, w_{i,N}\right]^T$ is the solution vector.

Noting that J_1 and M_1 is tri-diagonal matrix of order N given by:

$$J_1 = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}$$

$$M_1 = \begin{bmatrix} 1-2\theta & \theta & & & \\ \theta & 1-2\theta & \theta & & \\ & \ddots & \ddots & \ddots & \\ & & \theta & 1-2\theta & \theta \\ & & & \theta & 1-2\theta \end{bmatrix}$$

The constant vector b_i is of order N obtained from the boundary conditions are given by:

$$b_i = [\alpha_{2i-2} - \theta h^2 \alpha_{2i}, 0, \dots, 0, \beta_{2i-2} - \theta h^2 \beta_{2i}]^T \quad i = 1,2,\dots,m-1$$

now if $i=m$ we substitute equation (5.4) in (5.7) and obtain the system:

$$-w_{m,0} + 2w_{m,1} - w_{m,2} + h^2\left[\theta f(x_0, \alpha_0) + (1-2\theta) f(x_1, w_{1,1}) + \theta f(x_2, w_{1,2})\right] = 0$$

$$-w_{m,1} + 2w_{m,2} - w_{m,3} + h^2\left[\theta f(x_1, w_{1,1}) + (1-2\theta) f(x_2, w_{1,2}) + \theta f(x_3, w_{1,3})\right] = 0$$

$$\ddots \qquad \ddots \qquad \ddots$$

$$-w_{m,N-1} + 2w_{m,N} - w_{m,N+1} + h^2\left[\theta f(x_{N-1}, w_{1,N-1}) + (1-2\theta) f(x_N, w_{1,N}) + \theta f(x_{N+1}, \beta_0)\right] = 0$$

Which give the system:

$$\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \begin{bmatrix} w_{m,1} \\ w_{m,2} \\ \vdots \\ w_{m,N-1} \\ w_{m,N} \end{bmatrix} + \begin{bmatrix} 1-2\theta & \theta & & & \\ \theta & 1-2\theta & \theta & & \\ & \ddots & \ddots & \ddots & \\ & & \theta & 1-2\theta & \theta \\ & & & \theta & 1-2\theta \end{bmatrix} \begin{bmatrix} f(x_1, w_{1,1}) \\ f(x_2, w_{1,2}) \\ \vdots \\ f(x_{N-1}, w_{1,N-1}) \\ f(x_N, w_{1,N}) \end{bmatrix}$$

$$= \begin{bmatrix} \alpha_{m-2} - \theta h^2 f(w_0, y_{1,0}) \\ 0 \\ \vdots \\ 0 \\ \beta_{m-2} - \theta h^2 f(x_{N+1}, w_{1,N+1}) \end{bmatrix}$$

this system can be summarized on this formula:

$$J_1 W_m + h^2 M_1 f(x, W_1) = b_m \qquad \ldots(5.9)$$

The constant vectors b_m is of order N obtained from the boundary conditions are given by:

$$b_m = \left[\alpha_m - \theta h^2 f(x_0, w_{1,0}), 0, \ldots, 0, \beta_m - \theta h^2 f(x_{N+1}, w_{1,N+1})\right]^T$$

and the vector f of order N has the form: $\quad f = [f_1, f_2, \ldots, f_N]^T$

So we will solve these systems:

$$J_1 W_m + h^2 M_1 f(x, W_1) = b_m \qquad (5.9)$$

$$J_1 W_i + h^2 M_1 W_{i+1} = b_i \ , i = m-1, \ldots, 1 \qquad (5.8)$$

Using backward substitution for (5.8) and (5.9) and combined to give:

$$J_1 W_1 + (-1)^{m+1} h^{2m} M_1 (J_1^{-1} M_1)^{m-1} f(x, W_1) = b \qquad (5.10)$$

Where the constant vector b has the form:

$$b = \sum_{i=0}^{m-1} (-1)^i h^{2i} (M_1 J_1^{-1})^i b_{i+1}$$

Local Truncation Error:

To find the local truncation error for the previous method we will find the truncation error for equation first part of (5.7) as below:

$$w_{n-1} = w_n - hw_n' + \tfrac{h^2}{2} w_n'' - \tfrac{h^3}{6} w_n^{(3)} + \tfrac{h^4}{24} w_n^{(iv)} - \tfrac{h^5}{120} w_n^{(v)} + \tfrac{h^6}{360} w_n^{vi} + O(h^8) \tag{1}$$

$$w_{n+1} = w_n + hw_n' + \tfrac{h^2}{2} w_n'' + \tfrac{h^3}{6} w_n^{(3)} + \tfrac{h^4}{24} w_n^{(iv)} + \tfrac{h^3}{120} w_n^{(v)} + \tfrac{h^6}{360} w_n^{vi} + O(h^8) \tag{2}$$

by subtracting equations (1) & (2) from $2w_n$, we get the following equation

$$-w_{n-1} + 2w_n - w_{n+1} = -\tfrac{1}{12} h^4 w_n^{(iv)} - \tfrac{1}{360} h^6 w_n^{(vi)} + O(h^8) \tag{3}$$

now we will find the truncation error for the second part of equation (5.7)

$$\theta w_{n-1}'' = \theta w_n'' - \theta h w_n^{(3)} + \tfrac{\theta h^2}{2} w_n^{(iv)} - \tfrac{\theta h^3}{6} w_n^{(v)} + \tfrac{\theta h^4}{24} w_n^{vi} + O(h^8) \tag{4}$$

$$\theta w_{n+1}'' = \theta w_n'' + \theta h w_n^{(3)} + \tfrac{\theta h^2}{2} w_n^{(iv)} + \tfrac{\theta h^3}{6} w_n^{(v)} + \tfrac{\theta h^4}{24} w_n^{vi} + O(h^8) \tag{5}$$

by adding equations (4) & (5) to $(1-2\theta)w_n''$, we get the following equation

$$h^2\left(\theta w_{n-1}'' + (1-2\theta)w_n'' + \theta w_{n+1}''\right) = \theta h^4 w_n^{(iv)} + \tfrac{\theta}{12} h^6 w_n^{(vi)} + O(h^8) \tag{6}$$

by adding equation (3) to (4) we get the truncation error for (5.7)

$$\tau_n = \left(\theta - \tfrac{1}{12}\right) h^4 w_n^{(iv)} + \left(\tfrac{\theta}{12} - \tfrac{1}{360}\right) h^6 w_n^{(vi)} + O(h^8) \tag{5.11}$$

so from equation (5.11) we note that the method will be second-order convergent for $\theta = 0$ and fourth-order convergent for $\theta = \frac{1}{12}$.

Example: The Eighth-order boundary value problem

$$y^{(viii)}(x) = 7! \times e^{-8y(x)} - 2 \times 7! \times (1+x)^{-8} \quad ,0 < x < e^{1/2} - 1$$

With boundary conditions:

$$y(0) = 0 \qquad , \qquad y\left(e^{1/2} - 1\right) = \frac{1}{2}$$

$$y^{2i}(0) = -(2i-1)! \ , \qquad y^{2i}\left(e^{1/2} - 1\right) = y^{2i}(0)e^{-i} \ , \ \ i = 1,2,3$$

Has the exact solution is:

$$y^{*}(x) = \ln(1+x)$$

Using previous steps in the equation give the system:

$$y_2(x_n) = y_1''(x_n) \qquad , \quad \alpha_0 = 1 \, , \qquad \beta_0 = \frac{1}{2} \quad \ldots \ldots ..(1)$$

$$y_3(x_n) = y_2''(x_n) \quad , \quad \alpha_2 = -1 \, , \qquad \beta_2 = -e^{-1} \quad \ldots \ldots ..(2)$$

$$y_4(x_n) = y_3''(x_n) \quad , \quad \alpha_4 = -6 \, , \qquad \beta_4 = -6e^{-2} \quad \ldots \ldots ..(3)$$

$$y_4''(x_n) = 7! \times e^{-8y_1(x_n)} - 2 \times 7! \times (1+x_n)^{-8}$$

$$\alpha_6 = -120, \ \ \beta_6 = -120e^{-3} \qquad \ldots \ldots ..(4)$$

and by applying algorithm (5.1) using MATLAB see Chapter 6 (p.157-161), with *N=7* we get the following results as in Table 5.1 with acceptable tolerance value 1×10^{-5} that was accomplished with five iterations at $\theta = 0$

Table (5.1)

x_i	w_i	y_i	$\lvert w_i - y_i \rvert$
0	0	0	
0.08109	0.07769	0.07797	6.5699e-007
0.16218	0.15031	0.1503	8.307e-006
0.24327	0.21776	0.21775	1.8716e-005
0.32436	0.28096	0.28093	2.549e-005
0.40545	0.34038	0.34036	2.621e-005
0.48654	0.39647	0.39645	2.0766e-005
0.56763	0.44958	0.44957	1.0957e-005
0.64872	0.5	0.5	

5.2 Finite – Difference Method (5.1) for Solving Special Eighth-Order Non-linear Boundary Value Problems:

On this method we will directly replace the derivatives in the given differential equation by its finite difference approximations.

A Special Eight-Order Boundary -Value Problem:

Consider the special nonlinear eighth-order boundary value problem

$$y^{(viii)} = f(x, y) \quad , \qquad a \le x \le b \qquad , a, b, x \in \mathbb{R} \qquad \ldots (5.12)$$

With the associated boundary conditions:

$$y(a) = \alpha_0 \;, \quad y^{(ii)}(a) = \alpha_1 \,, \quad y^{(iv)}(a) = \alpha_2 \,, \quad y^{(vi)}(a) = \alpha_3$$

$$y(b) = \beta_0 \;, \quad y^{(ii)}(b) = \beta_1 \,, \quad y^{(iv)}(b) = \beta_2 \,, \quad y^{(vi)}(b) = \beta_3 \qquad \ldots (5.13)$$

We assume that f is real and differentiable as many times as required for $x\in$[a.b], and that the values of even derivatives at end points are real and finite constants.

We divide [a,b] into (N+1) equal subintervals whose endpoints are at $x_n = a + nh$, for n=0,1,…,N+1,$h = \frac{b-a}{N+1}$ where N≥7 is an integer. Assuming that the exact solution has abounded tenth derivative allows us to replace $y^{(viii)}$in the equation.

$$y^{(viii)}(x_i) = f(x_i, y(x_i))$$

The centered-finite difference formula for eighth-order differential equation which is obtained using undetermined coefficients method is:

$$y_i^{(viii)} = \frac{y_{i-4}-8y_{i-3}+28y_{i-2}-56y_{i-1}+70y_i-56y_{i+1}+28y_{i+2}-8y_{i+3}+y_{i+4}}{h^8} + \frac{h^2}{3}y^{(x)}(\xi) \quad \ldots(5.14)$$

By using equation (5.14) and substituting it in equation (5.12) we get

$$y_{i-4} - 8y_{i-3} + 28y_{i-2} - 56y_{i-1} + 70y_i - 56y_{i+1} + 28y_{i+2} - 8y_{i+3} + y_{i+4} - h^8 f_n + t_n = 0 \quad \ldots(5.15)$$

Where $y_n = y(x_n), f_n = f(x_n, y_n)$, t_n is the local truncation error at the point x_n, for n=1,2,3,…,N.

Now a finite difference method with truncation error of order $O(h^2)$ results by using this equation together with boundary conditions $y^{(2i)}(a) = \alpha_i$, $y^{(2i)}(b) = \beta_i$, $i = 0,1,2,3$, to define:

$$w_0^{(2i)} = \alpha_i \quad , \quad w_{N+1}^{(2i)} = \beta_{2i} , \quad i = 0,1,2,3$$

$$w_{i-4} - 8w_{i-3} + 28w_{i-2} - 56w_{i-1} + 70w_i - 56w_{i+1} + 28w_{i+2} - 8w_{i+3} + w_{i+4} - h^8 f_n = 0 \quad \ldots(5.16)$$

Where $w(x)$is the approximated solution for the problem and we will use the notation w_n to denote the solution at x_n. But we note that formula (5.16) can be applied to *N*-6 mesh points x_n , $(n = 4,5, \ldots \ldots, N-3)$ only. So we must find another special formula that consists with the point's *n*=1, 2, 3, *N*-2, *N*-1, *N* in order to use powers of matrix J_1 , in previous section. These formulas are assumed to be of the forms:

$$42w_1 - 48w_2 + 27w_3 - 8w_4 + w_5 - \sum_{i=0}^{4} h^{2i} c_{1,i} w_0^{(2i)} - h^8 f_1 = 0 \quad .(5.17)$$

$$-48w_1 + 69w_2 - 56w_3 + 28w_4 - 8w_5 + w_6 - \sum_{i=0}^{4} h^{2i} c_{2,i} w_0^{(2i)} - h^8 f_2 = 0 \quad \ldots(5.18)$$

$$27w_1 - 56w_2 + 70w_3 - 56w_4 + 28w_5 - 8w_6 + w_7 - \sum_{i=0}^{4} h^{2i} c_{3,i} w_0^{(2i)} - h^8 f_3 = 0 \quad \ldots(5.19)$$

$$w_{N-6} - 8w_{N-5} + 28w_{N-4} - 56w_{N-3} + 70w_{N-2} - 56w_{N-1} + 27w_N - \sum_{i=0}^{4} h^{2i} c_{3,i} w_{N+1}^{(2i)} - h^8 f_{N-2} = 0 \quad \ldots(5.20)$$

$$w_{N-5} - 8w_{N-4} + 28w_{N-3} - 56w_{N-2} + 69w_{N-1} - 48w_N - \sum_{i=0}^{4} h^{2i} c_{2,i} w_{N+1}^{(2i)} - h^8 f_{N-2} = 0 \quad \ldots(5.21)$$

$$w_{N-4} - 8w_{N-3} + 27w_{N-2} - 48w_{N-1} + 42w_N - \sum_{i=0}^{4} h^{2i} c_{1,i} w_{N+1}^{(2i)} - h^8 f_{N-2} = 0 \quad \ldots(5.22)$$

Where the 15 parameters $c_{1,i}$, $c_{2,i}$ *and* $c_{3,i}$, $(i = 0,1,2,3,4)$ are chosen for the convenience of using powers of *J* in the convergence analysis, we can find the parameters using the undetermined coefficients method. See Algorithm (5.2) & full MATLAB program in Chapter (6) p.(161-167).

The parameters values using the Algorithm (5.2) are:

Table (5.2)

Coefficient	*i=0*	*i=1*	*i=2*	*i=3*	*i=4*
$c_{1,i}$	14	-5	$^{19}/_{12}$	$-^{49}/_{72}$	$^{1158}/_{5141}$
$c_{2,i}$	-14	4	$^{-2}/_{3}$	$^{-7}/_{45}$	$^{-31}/_{2520}$
$c_{3,i}$	6	-1	$^{-1}/_{12}$	$^{-1}/_{360}$	$^{1}/_{20160}$

So the resulting *N*×*N* nonlinear system obtained from this method of equations (5.16), (5.17), (5.18), (5.19), (5.20), (5.21), and (5.22) can be summarized as:

$$
\begin{bmatrix}
42 & -48 & 27 & -8 & 1 & & & & & & & & \\
-48 & 69 & -56 & 28 & -8 & 1 & & & & & 0 & & \\
27 & -56 & 70 & -56 & 28 & -8 & 1 & & & & & & \\
-8 & 28 & -56 & 70 & -56 & 28 & -8 & 1 & & & & & \\
1 & 8 & 28 & -56 & 70 & -56 & 28 & -8 & 1 & & & & \\
 & 1 & -8 & 28 & -56 & 70 & -56 & 28 & -8 & 1 & & & \\
 & & & \ddots & & \ddots & & \ddots & & \ddots & & & \\
 & & & & 1 & -8 & 28 & -56 & 70 & -56 & 28 & -8 & \\
 & 0 & & & & 1 & -8 & 28 & -56 & 70 & -56 & 27 & \\
 & & & & & & 1 & -8 & 28 & -56 & 69 & -48 & \\
 & & & & & & & 1 & -8 & 27 & -48 & 42 &
\end{bmatrix}
\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ \vdots \\ w_{N-1} \\ w_N \end{bmatrix}
-
h^8 \begin{bmatrix} f_1 \\ \\ \\ \\ \\ f_N \end{bmatrix}
=
\begin{bmatrix}
c_{10}\alpha_0 + c_{11}h^2\alpha_1 + c_{12}h^4\alpha_2 + c_{13}h^6\alpha_3 + c_{14}h^8\alpha_4 \\
c_{20}\alpha_0 + c_{21}h^2\alpha_1 + c_{22}h^4\alpha_2 + c_{23}h^6\alpha_3 + c_{24}h^8\alpha_4 \\
c_{30}\alpha_0 + c_{31}h^2\alpha_1 + c_{32}h^4\alpha_2 + c_{33}h^6\alpha_3 + c_{34}h^8\alpha_4 \\
-\alpha_0 \\
\\
\\
-\beta_0 \\
c_{30}\beta_0 + c_{31}h^2\beta_1 + c_{32}h^4\beta_2 + c_{33}h^6\beta_3 + c_{34}h^8\beta_4 \\
c_{20}\beta_0 + c_{21}h^2\beta_1 + c_{22}h^4\beta_2 + c_{23}h^6\beta_3 + c_{24}h^8\beta_4 \\
c_{10}\beta_0 + c_{11}h^2\beta_1 + c_{12}h^4\beta_2 + c_{13}h^6\beta_3 + c_{14}h^8\beta_4
\end{bmatrix}
\quad \ldots.(5.23)
$$

This system can be summarized algebraically on the form:

$$J_1^4 W - h^8 f(x, W) - b = 0 \qquad \text{......(5.24)}$$

Where J_1^4 is the fourth power of matrix J_1, $W = [w_1, w_2, \dots, w_N]^T$ is the solution vector, b is the constant vector of order N .

We will use Newton's method for nonlinear systems, discussed in [Bruden & Faires, p.613-614] to approximate the solution to this system.

Local Truncation Error:

According to the finite-difference approximation to eight-order problem:

$$\tau_n = C_{10} h^{10} w_n^{(x)} + C_{11} h^{11} w_n^{(xi)} + \cdots \qquad \text{... (5.25)}$$

But as there is symmetry in the equation so that the odd terms will be equal zero, so by using equation (5.14) we find C_{10} as:

$$C_{10} = \frac{2(4)^{10} - 2*8(3)^{10} + 2*28(2)^{10} - 2*56(1)^{10}}{10!} = \frac{1}{3}$$

from the above we can get:

$$\tau_n = \frac{1}{3} h^{10} w_n^{(x)} + O(h^{12}), \text{ for } n = 3,4,\dots, N-3 \qquad \text{...(5.26)}$$

Now for *n=1, 2, 3 , N-2, N-1, N* , we will use equations (5.17), (5.18), (5.19), (5.20), (5.21), (5.22) to obtain the local truncation error as follows and by using algorithm(5.2) :

$$\tau_1 = \frac{-48(1)^{10} + 27(2)^{10} - 8(3)^{10} + (4)^{10} + 14(-1)^{10} - 5(-1)^8 + \frac{19}{12}(-1)^6 - \frac{49}{72}(-1)^4 + \frac{1158}{5141}(-1)^2}{10!} h^{10} w_1^{(x)} + O(h^{11}) = \frac{127}{416} h^{10} w_1^{(x)} + O(h^{11}) \qquad \text{...(5.27)}$$

and by the same steps we can find the remnant truncation errors, which are:

$$\tau_2 = \frac{198}{595} h^{10} w_2^{(x)} + O(h^{11}) \qquad \ldots(5.28)$$

$$\tau_3 = \frac{201599}{604798} h^{10} w_3^{(x)} + O(h^{11}) \qquad \ldots(5.29)$$

$$\tau_{N-2} = \frac{201599}{604798} h^{10} w_{N-2}^{(x)} + O(h^{11} \qquad \ldots(5.30)$$

$$\tau_{N-1} = \frac{198}{595} h^{10} w_{N-1}^{(x)} + O(h^{11}) \qquad \ldots(5.31)$$

$$\tau_N = \frac{127}{416} h^{10} w_N^{(x)} + O(h^{11}) \qquad \ldots(5.32)$$

Now we will discuss the following example to illustrate the previous method steps:

Example: The Eighth-order boundary value problem

$$y^{(viii)}(x) = 7! \times e^{-8y(x)} - 2 \times 7! \times (1+x)^{-8} \quad , 0 < x < e^{1/2} - 1$$

With boundary conditions:

$$y(0) = 0 \qquad , \qquad y\left(e^{1/2} - 1\right) = \frac{1}{2}$$

$$y^{2i}(0) = -(2i-1)! \ , \qquad y^{2i}\left(e^{1/2} - 1\right) = y^{2i}(0)e^{-i} \ , \ \ i = 1,2,3$$

Has the exact solution is: $\quad y^*(x) = \ln(1+x)$

Using the algorithm (5.3) in Chapter (6) p.(167-171),with second order convergent for N=7 and tolerance value 1×10^{-5} that was accomplished with four iterations, giving the results in Table 5.3

Table (5.3)

x_i	w_i	y_i	$\lvert w_i - y_i \rvert$
0	0	0	
0.08109	0.077823	0.07797	1.473e-4
0.16218	0.15003	0.1503	2.703e-4
0.24327	0.2174	0.21775	3.501e-4
0.32436	0.28055	0.28093	3.754e-4
0.40545	0.34001	0.34036	3.439e-4
0.48654	0.39619	0.39645	2.615e-4
0.56763	0.44942	0.44957	1.409e-4
0.64874	0.5	0.5	

The previous method is modified to solve especial fourth and sixth –order boundary value problems by finding the coefficients automatically using Algorithm (1.1), (5.2) together with Algorithm (5.3) and get the results that we want. See the full MATLAB program (p.171-175)

5.3 Finite – Difference Method (5.2) For Solving General Eighth-Order Non-linear Boundary Value Problems:

In previous section we dealt with special nonlinear problems and replaced directly the finite difference approximations. But in this method will deal with general eighth-order boundary value problems.

Assume the general eighth-order nonlinear boundary value problem:

$$y^{(viii)} = f\left(x, y, y', y'', y''', y^{(iv)}, y^{(v)}, y^{(vi)}, y^{(vii)}\right) \quad , a \le x \le b \quad \ldots(5.33)$$

With special boundary conditions:

$$y^{(2i)}(a) = \alpha_i \quad , y^{(2i)}(b) = \beta_i \quad , i = 0,1,2,3$$

Where f is real and differential as many times as required on [a,b], and that α_i, β_i , $(i = 0,1,2,3)$ are real finite constant.

As in the previous section we first divide [a,b] into equal subintervals whose endpoints are at $x_n = a + nh$, for n=0,1,2,...,$N+1$

To find the finite difference formula for this general problem we will use appropriate centered-difference formula for each derivative found in the problem especially the even derivatives, which is given as:

$$y_n'' = \frac{y_{n-1}-2y_n+y_{n+1}}{h^2} - \frac{h^2}{12}y^{(iv)}(\xi) \quad \text{...(5.34)}$$

$$y_n^{(iv)} = \frac{y_{n-2}-4y_{n-1}+6y_n-4y_{n+1}+y_{n+2}}{h^4} - \frac{h^2}{6}y^{(vi)}(\xi) \quad \text{...(5.35)}$$

$$y_n^{(vi)} = \frac{y_{n-3}-6y_{n-2}+15y_{n-1}-20y_n+15y_{n+1}-6y_{n+2}+y_{n+3}}{h^6} - \frac{h^2}{4}y^{(viii)}(\xi) \quad \text{...(5.36)}$$

$$y_n^{(viii)} = \frac{y_{n-4}-8y_{n-3}+28y_{n-2}-56y_{n-1}+70y_n-56y_{n+1}+28y_{n+2}-8y_{n+3}+y_{n+4}}{h^8} - \frac{h^2}{3}y^{(x)}(\xi) \quad (5.37)$$

Where $y_n = y(x_n)$,for n=0, 1, 2,…, $N+1$, then we substitute the approximations in equation (5.33) and by omitting the error terms we get the general finite difference formula with its boundary conditions, to explain this method we will discuss the following example :

$$(D^2 - A^2 - p\sigma)[(D^2 - A^2 - \sigma)(D^2 - A^2) + TD^2]y(x) + RD^2(D^2 - A^2 - \sigma)y(x) - f(x,y) = 0 , \quad a \le x \le b \quad \text{...(5.38)}$$

With the boundary conditions of the form:

$$y^{(2i)}(a) = \alpha_i \quad , y^{(2i)}(b) = \beta_i \quad , i = 0,1,2,3$$

The equation (5.38) can be simplified to the form:

$$(D^8 - c_1D^6 + c_2D^4 - c_3D^2 + c_4I)y(x) - f(x,y) = 0 \quad \text{...(5.39)}$$

Where:

$c_1 = 4A^2 + (p+2)\sigma$

$c_2 = 6A^4 + 3(p+2)\sigma A^2 + (1+2p)\sigma^2 + T$

$c_3 = 4A^6 + 3(p+2)\sigma A^4 + 2(2p+1)\sigma^2 A^2 + p\sigma(\sigma^2 + T) + (T-R)A^2$

$c_4 = A^8 + (p+2)\sigma A^6 + (2p+1)\sigma^2 A^4 + [p\sigma^3 - R(A^2+\sigma)]A^2$

To solve this problem first we will substitute the equations (5.34), (5.35), (5.36) & (5.37) in the equation (5.38) to get the following finite difference formula:

$$w_{n-4} - (8 + c_1h^2)w_{n-3} + (28 + 6c_1h^2 + c_2h^4)w_{n-2} - (56 + 15c_1h^2 + 4c_2h^4 + c_3h^6)w_{n-1} + (70 + 20c_1h^2 + 6c_2h^4 + 2c_3h^6 + c_4h^8)w_n - (56 + 15c_1h^2 + 4c_2h^4 + c_3h^6)w_{n+1} + (28 + 6c_1h^2 + c_2h^4)w_{n+2} - (8 + c_1h^2)w_{n+3} + w_{n+4} - h^8 f_n = 0 \qquad \ldots(5.40)$$

With $w_0^{(2i)} = \alpha_i \;\; , w_{N+1}^{(2i)} = \beta_i \;\; , i = 0,1,2,3$, where w_n is the approximated solution at x_n.

The finite difference formula in (5.40) can be applied for n=4,5,…,N-3 only, so we need a special formulas for n=1,2,3,N-2,N-1,N, that suits the powers of matrix J_1 in (5.11), so we will illustrate the steps to find.

For n=1

$$w_1'' = \frac{w_0 - 2w_1 + w_2}{h^2} \qquad \ldots(5.41)$$

$$w_1^{(iv)} = \frac{5w_1 - 4w_2 + w_3 - 2w_0 + h^2 w_0'' + \frac{1}{12}h^4 w_0^{(iv)}}{h^4} \qquad \ldots(5.42)$$

$$w_1^{(vi)} = \frac{-14w_1 + 14w_2 - 6w_3 + w_4 + 5w_0 - 2h^2 w_0'' + \frac{5}{6}h^4 w_0^{(iv)} + \frac{29}{180}h^6 w_0^{(vi)}}{h^6} \qquad \ldots(5.43)$$

$$w_1^{(viii)}=\frac{42w_1-48w_2+27w_3-8w_4+w_5-14w_0+5h^2w_0''-\frac{19}{12}h^4w_0^{(iv)}+\frac{49}{72}h^6w_0^{(vi)}+\frac{1158}{5141}h^8w_0^{(viii)}}{h^8} \quad \ldots(5.44)$$

By substituting the previous four equations in Eq.(5.34) give:

$$(42+14c_1h^2+5c_2h^4+2c_3h^6+c_4h^8)w_1-(48+14c_1h^2+4c_2h^4+c_3h^6)w_2+(27+6c_1h^2+c_2h^4)w_3-(8+c_1h^2)w_4+d_1-h^8f_1=0 \quad \ldots(5.45)$$

Where $d_1=-(14+5c_1h^2+2c_2h^4+c_3h^6)\alpha_0+h^2(5+2c_1h^2+c_2h^4)\alpha_1-h^4\left(\frac{19}{12}+\frac{5}{6}c_1h^2-\frac{1}{12}c_2h^4\right)\alpha_2+h^6\left(\frac{49}{72}-\frac{29}{180}c_1h^2\right)\alpha_3+\frac{1158}{5141}h^8f_0$

For n=2

$$w_2''=\frac{w_1-2w_2+w_3}{h^2} \quad \ldots(5.46)$$

$$w_2^{(iv)}=\frac{-4w_1+6w_2-4w_3+w_4+w_0}{h^4} \quad \ldots(5.47)$$

$$w_2^{(vi)}=\frac{14w_1-20w_2+15w_3-6w_4+w_5-4w_0+h^2w_0''+\frac{1}{12}h^4w_0^{(iv)}+\frac{1}{360}h^6w_0^{(vi)}}{h^6} \quad \ldots(5.48)$$

$$w_2^{(viii)}=\frac{-48w_1+69w_2-56w_3+28w_4-8w_5+w_6+14w_0-4h^2w_0''+\frac{2}{3}h^4w_0^{(iv)}+\frac{7}{45}h^6w_0^{(vi)}+\frac{31}{2520}h^8w_0^{(viii)}}{h^8} \quad \ldots(5.49)$$

By substituting the previous four equations in (5.33) give:

$$-(48+14c_1h^2+4c_2h^4+c_3h^6)w_1+(69+20c_1h^2+6c_2h^4+2c_3h^6+c_4h^8)w_2-(56+15c_1h^2+4c_2h^4+c_3h^6)w_3+(28+6c_1h^2+c_2h^4)w_4-(8+c_1h^2)w_5+w_6+d_2-h^8f_2=0 \quad \ldots(5.50)$$

Where

$$d_2=(14+4c_1h^2+c_2h^4)\alpha_0-h^2(4+c_1h^2)\alpha_1+h^4\left(\frac{2}{3}-\frac{1}{12}c_1h^2\right)\alpha_2+h^6\left(\frac{7}{45}-\frac{1}{360}c_1h^2\right)\alpha_3+\frac{31}{514}h^8f_0$$

For n=3

$$w_3''=\frac{w_2-2w_3+w_4}{h^2} \quad \ldots(5.51)$$

$$w_3^{(iv)} = \frac{w_1-4w_2+6w_3-4w_4+w_5}{h^4} \qquad \ldots(5.52)$$

$$w_3^{(vi)} = \frac{-6w_1+15w_2-20w_3+15w_4-6w_5+w_6+w_0}{h^6} \qquad \ldots(5.53)$$

$$w_3^{(viii)} = \frac{27w_1-56w_2+70w_3-56w_4+28w_5-8w_6+w_7-6w_0+h^2w_0''+\frac{1}{12}h^4w_0^{(iv)}+\frac{1}{360}h^6w_0^{(vi)}+\frac{1}{20160}h^8w_0^{(viii)}}{h^8}$$

$$\ldots(5.54)$$

By substituting the previous four equations in (5.34) give:

$$(27+6c_1h^2+c_2h^4)w_1-(56+15c_1h^2+14c_2h^4+c_3h^6)w_2+(70+20c_1h^2+6c_2h^4+2c_3h^6+c_4h^8)w_3-(56+15c_1h^2+4c_2h^4+c_3h^6)w_4+(28+6c_1h^2+c_2h^4)w_5-(8+c_1h^2)w_6+w_7+d_3-h^8f_3=0 \qquad \ldots(5.55)$$

Where $d_3=-(6+c_1h^2)\alpha_0+h^2\alpha_1+\frac{1}{12}h^4\alpha_2+\frac{1}{360}h^6\alpha_3+\frac{1}{20160}h^8f_0$

For $n=N$-2

$$w_{N-2}'' = \frac{w_{N-3}-2w_{N-2}+w_{N-1}}{h^2} \qquad \ldots(5.56)$$

$$w_{N-2}^{(iv)} = \frac{w_{N-4}-4w_{N-3}+6w_{N-2}-4w_{N-1}+w_N}{h^4} \qquad \ldots(5.57)$$

$$w_{N-2}^{(vi)} = \frac{-6w_{N-4}+15w_{N-3}-20w_{N-2}+15w_{N-1}-6w_N+w_{N+1}}{h^6} \qquad \ldots(5.58)$$

$$w_{N-2}^{(viii)} =$$

$$\frac{w_{N-6}-8w_{N-5}+28w_{N-4}-56w_{N-3}+70w_{N-2}-56w_{N-1}+27w_N-6w_{N+1}+h^2w_{N+1}''+\frac{1}{12}h^4w_{N+1}^{(iv)}+\frac{1}{360}h^6w_{N+1}^{(vi)}+\frac{1}{20160}h^8w_{N+1}^{(viii)}}{h^8}$$

$$\ldots(5.59)$$

By substituting the previous four equations in (5.34) give:

$$(27+6c_1h^2+c_2h^4)w_N-(56+15c_1h^2+14c_2h^4+c_3h^6)w_{N-1}+(70+20c_1h^2+6c_2h^4+2c_3h^6+c_4h^8)w_{N-2}-(56+15c_1h^2+4c_2h^4+$$

$c_3h^6)w_{N-3} + (28 + 6c_1h^2 + c_2h^4)w_{N-4} - (8 + c_1h^2)w_{N-5} + w_{N-6} + d_{N-2} - h^8 f_{N-2} = 0$ …(5.60)

Where $d_{N-2} = -(6 + c_1h^2)\beta_0 + h^2\beta_1 + \frac{1}{12}h^4\beta_2 + \frac{1}{360}h^6\beta_3 + \frac{1}{20160}h^8 f_{N+1}$

For n=N-1

$$w''_{N-1} = \frac{w_{N-2} - 2w_{N-1} + w_N}{h^2} \quad \text{…(5.61)}$$

$$w^{(iv)}_{N-1} = \frac{w_{N-3} - 4w_{N-2} + 6w_{N-1} - 4w_N + w_{N+1}}{h^4} \quad \text{…(5.62)}$$

$$w^{(vi)}_{N-1} = \frac{w_4 - 6w_{N-3} + 15w_{N-2} - 20w_{N-1} + 14w_N - 4w_{N+1} + h^2 w''_{N+1} + \frac{1}{12}h^4 w^{(iv)}_{N+1} + \frac{1}{360}h^6 w^{(vi)}_{N+1}}{h^6}$$

…(5.63)

$$w^{(viii)}_{N-1} = \frac{w_{N-5} - 8w_{N-4} + 28w_{N-3} - 56w_{N-2} + 69w_{N-1} - 48w_N + 14w_{N+1} - 4h^2 w''_{N+1} + \frac{2}{3}h^4 w^{(iv)}_{N+1} + \frac{7}{45}h^6 w^{(vi)}_{N+1} + \frac{31}{2520}h^8 w^{(viii)}_{N+1}}{h^8}$$

…(5.64)

By substituting the previous four equations in (5.34) give:

$-(48 + 14c_1h^2 + 4c_2h^4 + c_3h^6)w_N + (69 + 20c_1h^2 + 6c_2h^4 + 2c_3h^6 + c_4h^8)w_{N-1} - (56 + 15c_1h^2 + 4c_2h^4 + c_3h^6)w_{N-2} + (28 + 6c_1h^2 + c_2h^4)w_{N-3} - (8 + c_1h^2)w_{N-4} + w_{N-5} + d_{N-1} - h^8 f_{N-1} = 0$ …(5.65)

Where

$$d_{N-1} = (14 + 4c_1h^2 + c_2h^4)\beta_0 - h^2(4 + c_1h^2)\beta_1 + h^4\left(\frac{2}{3} - \frac{1}{12}c_1h^2\right)\beta_2 + h^6\left(\frac{7}{45} - \frac{1}{360}c_1h^2\right)\beta_3 + \frac{31}{2520}h^8 f_{N+1}$$

For n=N

$$w''_N = \frac{w_{N-1} - 2w_N + w_{N+1}}{h^2} \quad \text{…(5.66)}$$

$$w_N^{(iv)} = \frac{w_{N-2}-4w_{N-1}+5w_N-2w_{N+1}+h^2w''_{N+1}+\frac{1}{12}h^4w_{N+1}^{(iv)}}{h^4} \qquad \ldots(5.67)$$

$$w_N^{(vi)} = \frac{w_{N-3}-6w_{N-2}+14w_{N-1}-14w_N+5w_{N+1}-2h^2w''_{N+1}+\frac{5}{6}h^4w_{N+1}^{(iv)}+\frac{29}{180}h^6w_{N+1}^{(vi)}}{h^6}$$

$$\ldots(5.68)$$

$$w_N^{(viii)} = \frac{w_{N-4}-8w_{N-3}+27w_{N-2}-48w_{N-1}+42w_N-14w_{N+1}+5h^2w''_{N+1}-\frac{19}{12}h^4w_{N+1}^{(iv)}+\frac{49}{72}h^6w_{N+1}^{(vi)}+\frac{1158}{1158}h^8w_{N+1}^{(viii)}}{h^8}$$

$$\ldots(5.69)$$

By substituting the previous four equations in (5.34) give:

$$(42+14c_1h^2+5c_2h^4+2c_3h^6+c_4h^8)w_N-(48+14c_1h^2+4c_2h^4+c_3h^6)w_{N-1}+(27+6c_1h^2+c_2h^4)w_{N-2}-(8+c_1h^2)w_{N-3}+d_N-h^8f_N=0 \qquad \ldots(5.70)$$

Where

$$d_N=-(14+5c_1h^2+2c_2h^4+c_3h^6)\beta_0+h^2(5+2c_1h^2+c_2h^4)\beta_1-h^4\left(\frac{19}{12}+\frac{5}{6}c_1h^2-\frac{1}{12}c_2h^4\right)\beta_2+h^6\left(\frac{49}{72}-\frac{29}{180}c_1h^2\right)\beta_3+\frac{1158}{5141}h^8f_N$$

Now the system of previous equations (5.40), (5.45), (5.51), (5.55), (5.60), (5.65), & (5.70) can be summarized as:

$$[J^4+c_1h^2J^2+c_2h^4J^4+c_3h^6J+c_4h^8I]W-h^8f(x,W)+d=0 \quad \ldots(5.71)$$

where $W=[w_1,w_2,\ldots,w_N]^T$ the solution vector , that was found by solving the nonlinear algebraic system (5.71) using Newton's method.

$$d=[d_1,d_2,d_3,-\alpha_0,0,\ldots,0,-\beta_0,d_{N-2},d_{N-1},d_N]^T$$

The Local Truncation Errors:

We can find easily the local truncation error for equation (5.40) by using equations (5.2), (5.3), (5.4) & (5.5) for *n*=4, 5, …, *N-3* as follows:

$$\tau_n = h^{10}\left[\tfrac{1}{3}w_n^{(x)} - \tfrac{1}{4}c_1 w_n^{(viii)} + \tfrac{1}{6}c_2 w_n^{(vi)} - \tfrac{1}{12}c_3 w_n^{(iv)}\right] + O(h^{12}) \quad \ldots(5.72)$$

but for *n*=1, 2, 3, *N*-2, *N*-1, *N* we will use the associated finite-difference approximations with their truncation errors that were used to find equations (5.13), (5.19), (5.24), (5.29), (5.34) &(5.39) by using (local Truncation) algorithm:

$$\tau_1 = h^{10}\left[\tfrac{127}{416}w_1^{(x)} - \tfrac{479}{2016}c_1 w_1^{(viii)} + \tfrac{59}{360}c_2 w_1^{(vi)} - \tfrac{1}{12}c_3 w_1^{(iv)}\right] + O(h^{11}) \quad \ldots (5.73)$$

$$\tau_2 = h^{10}\left[\tfrac{198}{595}w_2^{(x)} - \tfrac{1260}{5041}c_1 w_2^{(viii)} + \tfrac{1}{6}c_2 w_1^{(vi)} - \tfrac{1}{12}c_3 w_1^{(iv)}\right] + O(h^{11}) \quad \ldots (5.74)$$

$$\tau_3 = h^{10}\left[\tfrac{201599}{604798}w_3^{(x)} - \tfrac{1}{4}c_1 w_1^{(viii)} + \tfrac{1}{6}c_2 w_1^{(vi)} - \tfrac{1}{12}c_3 w_1^{(iv)}\right] + O(h^{11}) \quad \ldots(5.75)$$

$$\tau_{N-2} = h^{10}\left[\tfrac{201599}{604798}w_{N-2}^{(x)} - \tfrac{1}{4}c_1 w_{N-2}^{(viii)} + \tfrac{1}{6}c_2 w_{N-2}^{(vi)} - \tfrac{1}{12}c_3 w_{N-2}^{(iv)}\right] + O(h^{11}) \quad \ldots(5.76)$$

$$\tau_{N-1} = h^{10}\left[\tfrac{198}{595}w_{N-1}^{(x)} - \tfrac{1260}{5041}c_1 w_{N-1}^{(viii)} + \tfrac{1}{6}c_2 w_{N-1}^{(vi)} - \tfrac{1}{12}c_3 w_{N-1}^{(iv)}\right] + O(h^{11}) \quad \ldots(5.77)$$

$$\tau_N = h^{10}\left[\tfrac{127}{416}w_N^{(x)} - \tfrac{479}{2016}c_1 w_N^{(viii)} + \tfrac{59}{360}c_2 w_N^{(vi)} - \tfrac{1}{12}c_3 w_N^{(iv)}\right] + O(h^{11}) \quad \ldots(5.78)$$

This method is like method (4.6) with some differences in coefficients and matrix diagonal that we can see some of their results in reference [3].

CHAPTER SIX

Difficulties, Conclusions and Computer Programming

This chapter discusses the difficulties that we faced while preparing this thesis, and how we dealt with, and the associated MATLAB 7.0 computer programs we constructed for this aim. Besides that the development of some new programs to reduce time and efforts in calculations, and others to solve linear boundary-value problems of order three up to seven.

6.1 Difficulties:

We faced many difficulties while preparing this thesis that can be summarized in these questions:

(i) How can we write a derivative of order (O) using any set of points $w(x+ih)$ easily with simple time and effort?

(ii) How can we write a derivative of order (O) using any set of points $w(x+ih)$ and any set of derivatives we want, easily with simple time and effort?

(iii) How can we write an even derivative of order (O) using a set of especial points $w(x+ih)$ and a set of even derivatives equal and less than its order, easily with less time and effort?

(iv) How can we use the powers of $J1$ matrix entries without using the power process?

(v) How can we find the local truncation error for any finite difference approximation, easily with simple time and effort?

(vi) Method (5.2) approximates the solution only for eighth-order BVPs with $N>7$, consequently how we can modify this method to approximate the solution for some Even-order BVPs with $N\geq O-1$?

6.2 Conclusions:

The approximated solutions for problems that we discussed in this thesis have good errors using finite-difference methods, but they depend on the problem order, as the order is smaller the results will be better.

The main reasons for accumulative errors in each method depend on using more derivative approximations in the high-order problems, and also on the boundary conditions for the problem.

We note that when α_0 or β_0 absence the errors in approximation are large and method needs to use higher order of accuracy at the endpoints.

The seventh-order problem in chapter four takes long time in testing and calculating to get more accurate results. But we cannot get good results.

The Finite-Difference methods need hard-work while deriving the finite-difference approximations especially for these high-order problems and using many permutations at their endpoints to get our convergence order, but in this thesis we simplify this routine by using especial and new algorithms.

6.3 Computer Programs:

In this section we use (MATLAB 7.0) package in programming and we use MATLAB functions to simplify our work.

Now we will discuss some important computer programs that we did not discuss in the previous chapters that solve some of above difficulties and others that modified and developed especially in chapter five.

Program (5.1):

This program is used to approximate the solution for Even-BVPs by transforming it to system of second- order problem. This method is second order convergent if $\theta = 0$ and fourth-order convergent if $\theta = 1/12$ and this was modified to select any θ the user want.

The Finite-Difference Method using 2m-order BVPs Program

```
% To Approximate The Solution for Non-linear Even-Order BVPs.

% With Boundary Conditions d(2i)W0 ,d(2i)WN+1.

% INPUTS: - The Order of the Problem

%         - The Function for BVP f(X,W)

%         - The Derivative of f(X,W) with respect to W

%         - The Interval End-points

%         - The Number of Subinterval

%         - The Boundary Condition for problem
```

```
%        - The Maximum Number of Iterations

%        - The Tolerance

%        - Theta

% OUTPUTS: - The Approximation of W(X)

syms ('X','W');

o=input ('Write the order of problem=');

a=input ('Write the initial point of interval a =');

b=input ('Write the end point of interval b =');

N=input ('Write the number of sub-interval [a,b] =');

display ('Write the boundary condition as vector [W0 d2iW0] =');

A0=input (' ');

display ('Write the boundary condition as vector [WN+1 d2iWN+1] =');

B0=input (' ');

MM=input ('Write the maximum number of iteration=');

F3=input ('Write the exact function for the problem=');

T=input ('Write the tolerance you want=');

Ta0=input ('Write 0<=Theta <=1, Theta=');

% Calculate the step-size
```

```
h= (b-a)/ (N+1); m=o/2;

% The initial approximation for Wi

for i=1:N

    x(i)=a+i*h; w(i)=A0(1)+i*h*((A0(1)-B0(1))/(b-a));end

%The appropriate matrix for system

for i=1:N

    for j=1:N

    if i==j     J1(i,j)=2,    M1(i,j)=(1-2*Ta0);

% The constants vector for problem

for i=1:m+1

for j=1:N

if j==1          d1(i,j)=A0(i);

elseif j==N   d1(i,j)=B0(i);        end , end, end

d=d1(1,1:N)-(Ta0*h^2)*d1(2,1:N);

for i=2:m

d11=((M1*(Jv1*M1)^(i-1))*(d1(i,1:N)-
(Ta0*h^2)*d1(i+1,1:N))')*(h^(2*(i-1)))*(-1)^(i-1);

d=d+d11';end

k=1;
```

```
while (k<=MM)

% Evaluate the function and its derivative at W

Fn1=FmethodII(x,w,F);

Fn2=F2methodII(x,w,F2);

Fn3=ExactFn(x,F3);

% The Jacobian Matrix for system

c=((-1)^(m-1))*(h^(2*m));J2=c*(M1*((Jv1*M1)^(m-1)))*(diag(c*Fn2));

J=J1+J2;

% The system function at Wi

f=-(J1*w'+(c*(M1*((Jv1*M1)^(m-1)))*Fn1')-d');

% The solution vector for system Jv=-f

v=linsolve(J,f);

if t<=T       break, end

% The new approximated solution for system

w=w+v';  x;   ww(k,1:N)=w;

k=k+1;  end

if o==0    display('The Method Is Second-Order Convergent');

elseif o==1/12  display('The Method Is Fourth-Order Convergent'); end
```

```
fprintf(1,'   x                ww                exact\n'); [x'  w' Fn3' abs(w-Fn3)']
```

Program (5.2):

This program is designed in order to solve difficulty (iii). While we construct a general formula for solving eighth-order BVPs, we need a special formula for six points using especial even-derivatives as boundary conditions and set of points W(x+ih) which their coefficients are the entries of the power of J_1.

This program find the coefficients of the even derivatives less than or equal to the problem order (O).

The program steps:

(i) Find the matrix of h coefficients for w(x+ih) points we enter.

(ii) Determine the index of W_n at which we want to use the associated row of $J_1^{O/2}$.

(iii) Find the matrix of h coefficients for even derivatives $w_0^{(2i)}$ less than or equal to the problem order.

(iv) Find the constant vector, which came from subtracting two parts:

1. by multiplying the results from step(i) and step(2).
2. the corresponding problem order factorial.

(v) Use forward substitution method to solve (iii) and (iv).

```
% INPUTS: - The Order of the Derivative.

%         - The Number & Coefficient of h for W(x+ah).

%         - The Number & Even-Derivative & Coefficient of h for
DW(x+ah).
```

```
% OUTPUTS: - The Coefficients For W(x+ah).
%          - The Coefficients for Even-DW(x+ah).
format rat
o=input('The Order of Differential Equation =');
n=input('The xn index =');
N1=input('Number of Wn terms =');
aN1=input('The coefficient of h for w(x+ah)as vector between [  ]  =');
N2=input('Number of even derivatives =');
aN2=input('The coefficient of h for W0& Dw(x+ah)as vector between [  ]  =');
N=N1+N2;
%To determine the h power associated with even derivatives
h=sym('h','unreal'),c=1/(h^o);
for i=1:N2
dc(i)=h^((i-1)*2);end
% The system of the coefficients of h for W(x+nh)
for i=1:N1+N2+1
   for j=1:N1
      if (i==1)       aaN1(i,j)=1;
```

```
        else          aaN1(i,j)=aN1(j)^(i-1);    end , end, end

% The system of the coefficients of h for even derivatives D2iW(x+nh)

%lower triangular matrix

aaN2(1,1)=1;

for i=1:N2+1

    for j=1:N2

        if j==1      aaN2(i,j)=aN2(j)^(2*(i-1));

        elseif i==1 & j>1        aaN2(i,j)=0;

        elseif i==j   aaN2(i,j)=factorial((2*(i-1)));

        elseif j<I

 aaN2(i,j)=(aN2(j)^(2*abs(i-j)))*(factorial((2*(i-1)))/factorial(2*abs(i-j)));

         end, end ,end

%The constant vector for the system of unknown Au=b.

fo=factorial(o);

for i=1:N2

    if i==(o/2+1)      bN2(i)=fo;

    else bN2(i)=0; end, end

% The constant vector for Ad2iW=b that is the result from multiplying
aaN1 in the suitable vector of J1
```

```
JJ1=coefn2(o,N1,n,N);    bN21(1)=aaN1(1,1:N1)*JJ1';

  for i=2:N2

  bN21(i)=(aaN1((2*i-1),1:N1))*JJ1';   end

  b=bN2-bN21,bb=(b);

% use forward substitution method for solving Aw=b

cc1=forwmethod(N2,b,aaN2);

fprintf('The Coefficient for W0 & Even-Derivatives\n');cc1

fprintf('Multiplying By:\n');dc

%The local truncation error which was given from to part t1 from Wn and
t2 from even derivatives d2iW

t1=aaN1(o+3,1:N1)*JJ1';  t2=aaN2(N2+1,1:N2)*cc1';
t=(t1+t2)/factorial(o+2);

fprintf('The Local Truncation Error for Wn\n',t) ,t

fprintf('Multiplying by:\n'); h^(o+2)
```

The [coefn2] Programming Function:

This is a programming function used to find the entries of O/2 rows of J_1 to the power O/2 matrix that is necessary for finite difference method 5.1, method 5.2, and method 5.3 for solving even-order BVPs with even-boundary conditions.

We note that while programming previous methods, the J_1 powers entries are used many times to determine the finite-difference formula for the problem. When we divide the problem interval into *N+1* subintervals, we used $J_1$of order *N×N*. If we need $J_1^{o/2}$ the computer will multiplying J_1 (o/2)times, which needs *(2N×o/2)* processes for only one entry. So we think of a code that gives the entries without using the power of J_1.

The main steps:

(i) Find the central finite-difference approximation for problem.

(ii) Use the central finite-difference approximation to find the matrix rows that are not similar to the central formula.

(iii) Omit first entries respectively down –up and subtracting it from the following entry.

Example: Find the J_1^4matrix entries, which are needed for solving eight-order BVPs.

We note that middle matrix rows are similar to the central finite-difference approximation for eighth-order problem (O=8) as:

$$
\begin{array}{r}
r1 \\ r2 \\ r3 \\ r4 \\ r5 \\ \\ \\ rN-3 \\ rN-2 \\ rN-1 \\ rN
\end{array}
\begin{bmatrix}
42 & -48 & 27 & -8 & 1 & & & & & & & \\
-48 & 69 & -56 & 28 & -8 & 1 & & & & & & \\
27 & -56 & 70 & -56 & 28 & -8 & 1 & & & & 0 & \\
-8 & 28 & -56 & 70 & -56 & 28 & -8 & 1 & & & & \\
\color{red}{1} & \color{red}{8} & \color{red}{28} & \color{red}{-56} & \color{red}{70} & \color{red}{-56} & \color{red}{28} & \color{red}{-8} & \color{red}{1} & & & \\
 & 1 & -8 & 28 & -56 & 70 & -56 & 28 & -8 & 1 & & \\
 & & & & \ddots & & \ddots & & \ddots & & \ddots & \\
 & & & & 1 & -8 & 28 & -56 & 70 & -56 & 28 & -8 \\
 & 0 & & & & 1 & -8 & 28 & -56 & 70 & -56 & 27 \\
 & & & & & & 1 & -8 & 28 & -56 & 69 & -48 \\
 & & & & & & & 1 & -8 & 27 & -48 & 42
\end{bmatrix}
$$

Note that the row (5=1+O/2) is the entry of the central finite difference formula for (O=8), and to find the upper rows we follow these steps:

row (4): results from omitting the first entry which is (1) from row(5)

$$r(4) = -8 \quad 28 \quad -56 \quad 70 \quad -56 \quad 28 \quad -8 \quad 1 \quad 0$$

row(3): results from omitting the first entry which is (-8) from the previous row(4) and subtracting the previous omitting entry from the new first entry of row(3).

$$r(3) = (28 - 1) \quad -56 \quad 70 \quad -56 \quad 28 \quad -8 \quad 1 \quad 0 \quad 0$$

$$r(3) = 27 \quad -56 \quad 70 \quad -56 \quad 28 \quad -8 \quad 1 \quad 0 \quad 0$$

row(2):results from omitting the first entry which is (27) from row(3) and subtracting the previous omitting entry -8 ,1 from the first two new entry of row(2).

$$r(2) = (-56 + 8) \quad (70 - 1) \quad -56 \quad 28 \quad -8 \quad 1 \quad 0 \quad 0 \quad 0$$

$$r(2) = -48 \quad 69 \quad -56 \quad 28 \quad -8 \quad 1 \quad 0 \quad 0 \quad 0$$

Row(1):results from the same routine which is used above.

$$r(1) = (69 - 27) \quad (-56 + 8) \quad (28 - 1) \quad -8 \quad 1 \quad 0 \quad 0 \quad 0$$

$$r(1) = 42 \quad -48 \quad 27 \quad -8 \quad 1 \quad 0 \quad 0 \quad 0$$

Now because of symmetry, the lower O/2 rows are the same as the above rows with some modifications by using especial MATLAB functions to calculate the reverse of vectors and flip matrices up-down, which are consequently (seqreverse , flipud). And remaining rows entries are central finite difference formula and we determine the position of zeros according to the size of matrix, but this algorithm must use matrix of size greater than or equal to (O-1).

We note from these steps that we do not need for using the power matrix, only what we need is the finite difference formula for the problem and the matrix size.

Program (5.3):

This method is used to approximate the solution for Eighth-Order BVPs with Even-boundary conditions.

$$y^{(viii)} = f(x, y) \quad , \qquad a \leq x \leq b \qquad , a, b, x \in \mathbb{R}$$

with boundary conditions $\quad y_0^{(2i)} = \alpha_i \quad , \; y_{N+1}^{(2i)} = \beta_{2i} \, , \; i = 0,1,2,3$

Noting that N is greater than problem order.

% The Second-Order Finite-Difference Method 5.1

% To Approximate The Solution for Non-linear Eighth-Order BVP.

% With Boundary Conditions d(2i)W0 ,d(2i)WN+1.

% INPUTS:- The Function For BVP f(X,W)

% - The Derivative of f(X,W) with respect to W

% - The Interval End-points

% - The Number of Subinterval

% - The Boundary Condition for problem

```
%        - The Maximum Number of Iterations

%        - The Tolerance

% OUTPUTS: The Approximated Solution for The 8th-order BVP

syms('X','W');

F=input('Write The F(xn,wn)for Problem:');

F2=input('Write the derivative for F(xn,wn) with respect to Wn:');

a=input('Write the initial point of interval a =');

b=input('Write the end point of interval b =');

N=input('Write the number of sub-interval[a,b] =');

A0=input('Write The boundary condition as vector[W0 d2W0 d4W0 d6W0
d8W0]=');

B0=input('Write The boundary condition as vector[WN+1 d2WN+1
d4WN+1 d6WN+1 d8WN+1]=');

M=input('Write The Maximum Number of iteration=') ;

F3=input('Write The Exact Function for The Problem=');

T=input('Write The Tolerance You want=');

% Calculate the step-size

h=(b-a)/(N+1);

% The Initial Approximation for Wi
```

```
for i=1:N

   x(i)=a+i*h;  w(i)=A0(1)+i*h*((A0(1)-B0(1))/(b-a));end

k=1

while (k<=M)

% The constants vector for problem

if (N==7)

d1=[14 -5*h^2 19*h^4/12 -49*h^6/72 1158*h^8/5141;-14 4*h^2 -2*h^4/3
-7*h^6/45 -31*h^8/2520;6 -1*h^2 -1*h^4/12 -1*h^6/360 1*h^8/20160] ;

dN=[6 -1*h^2 -1*h^4/12 -1*h^6/360 1*h^8/20160;-14 4*h^2 -2*h^4/3 -
7*h^6/45 -31*h^8/2520;14 -5*h^2 19*h^4/12 -49*h^6/72 1158*h^8/5141];

d=[(d1*A0')' -(A0(1)+B0(1)) (dN*B0')'];

else

d1=[14 -5*h^2 19*h^4/12 -49*h^6/72 1158*h^8/5141;-14 4*h^2 -2*h^4/3
-7*h^6/45 -31*h^8/2520;6 -1*h^2 -1*h^4/12 -1*h^6/360 1*h^8/20160;-1 0
0 0 0] ;

dN=[-1 0 0 0 0;6 -1*h^2 -1*h^4/12 -1*h^6/360 1*h^8/20160;-14 4*h^2 -
2*h^4/3 -7*h^6/45 -31*h^8/2520;14 -5*h^2 19*h^4/12 -49*h^6/72
1158*h^8/5141];

d=[(d1*A0')' zeros(1,N-8) (dN*B0')'];end

% Evaluate The function and its derivative at W
```

```
Fn1=FmethodII(x,w,F);

Fn2=F2methodII(x,w,F2);

Fn3=ExactFn(x,F3);

%The Appropriate Matrix for system

for i=1:N

    for j=1:N

    if i==j     J1(i,j)=-2;

    elseif abs(i-j)==1          J1(i,j)=1;    end, end, end

% The Jacobian Matrix for system

J=J1^4-diag(h^8*Fn2);

% The system function at Wi

f=-(J1^4*w'-h^8*Fn1'-d');

% The solution vector for system Jv=-f

v=linsolve(J,f);    t=norm(v);

if t<=T    break, end

% The New Approximated solution for system

w=w+v';  ww(k,1:N)=w;

k=k+1; end
```

disp(' The x and it is approximated solution is:');[x' ww'];

Modified Program (5.3):

This program is modification for program(5.3) that is modified to approximate the solution for any Even-Order BVPs with Even-boundary conditions less than or equal to the order of the problem and with N≥ O-1 equal subsets .

The modification is depending on usingFN1, FN3, coefn, coefn2 and FN4 programming functions.

coefn and coefn2 programming functions we discuss previously so now we will discuss FN1,FN3and FN4 then the method.

%This function is used to calculate constant vector for any even-order BVPs

FN1 -Function

```
function [res]=FN1(o,N,h,A0,B0)

Ho(1)=1;

for i=2:1+o/2

   Ho(i)=h^(2*(i-1)),end

if N>o-1  d1=-FN3(o,N)*diag(Ho);   dN=flipud(d1)*diag(Ho);

   d11=[(d1*A0')' zeros(1,N-4) (dN*B0')'];    res=d11;

elseif N==o-1  d1=-FN3(o,N)*diag(Ho(2:o/2);
dN=flipud(d1)*diag(Ho(2:o/2);
```

```
d11=[(d1*A0')' -(A0(1)+B0(1)) (dN*B0')'];   res=d11;

else fprintf(1,'Sorry The Program cannot do this Presses'); end
```

FN2 –Function:

```
%This function is used to find Even-derivative coefficients for any even-order BVPs

function [res]=FN3(o,N)

%N2 :the number of row different on the power of J1

N2=1+o/2;

if N>o-1    kk=o/2

elseif N==o-1  kk=o/2-1 ,end

for k=1:kk

n=k;  N1=k+o/2;

% The coefficient vector for h in W(x+ah)  point according to J1 power

aN1=[-k+1:o/2];

% The coefficient vector for h in even-derivative less than and equal to O d2iW(x+ah)

aN2=-k*ones(1,N2); fo=factorial(o);

% The first part of constant vector to find the even derivatives coefficients

for i=1:N2
```

```
   if i==(o/2+1)         bN2(i)=fo;
   else bN2(i)=0;    end , end
% The coefficients matrix of h power for the set point W(x+ah)
for i=1:N1+N2
   for j=1:N1
 if (i==1)        aaN1(i,j)=1;
else           aaN1(i,j)=aN1(j)^(i-1);   end , end ,end
% The coefficients matrix of h power for the set of even derivative d2i
W(x+ah)
aaN2(1,1)=1;
for i=1:N2
   for j=1:N2
 if j==1   aaN2(i,j)=aN2(j)^(2*(i-1));
elseif i==1 & j>1     aaN2(i,j)=0;
elseif i==j    aaN2(i,j)=factorial((2*(i-1)));
 elseif j<i
aaN2(i,j)=(aN2(j)^(2*abs(i-j)))*(factorial((2*(i-1)))/factorial(2*abs(i-j)));
end, end, end
% The coefficients vector associated with J1 power rows
```

```
JJ1=coefn2(o,N1,n,N);
%The second part of constant vector
bN21(1)=aaN1(1,1:N1)*JJ1';
  for i=2:N2
      bN21(i)=(aaN1((2*i-1),1:N1))*JJ1';   end
b=bN2-bN21;   bb=floor(b);
% The even derivative coefficients
cc1=forwmethod(N2,b,aaN2);
ch(k,1:N2)=cc1; end; res=ch;
```

FN4 - Function :

```
% This function to determine J1^O/2  entries
function [rec]=FN4(o,N)
N2=1+o/2; NN=o+1;
% To get the central finite-difference formula (The middle rows of J1^O/2)
Jm1=coefn(o,NN);
if N>o-1
for i=1:o/2
n=i;  N1=i+o/2;  aN1=[-i+1:o/2];  aN2=-i*ones(1,N2);
```

```
% To get the entry for  O/2 rows of J1 to power O/2
Jm=coefn2(o,N1,n);   E=seqreverse(Jm);
A(i,1:N)=[Jm zeros(1,N-N1)]; A(N-i+1,1:N)=[zeros(1,N-N1) E]; end
for i=N2:N-o/2
   A(i,1:N)=[zeros(1,i-N2) Jm1 zeros(1,(N-NN-i+N2))];end
% The J1 to power o/2 if N=o-1
elseif N==o-1
for i=1:o/2-1 n=i;N1=i+o/2; aN1=[-i+1:o/2]; aN2=-i*ones(1,N2);
Jm=coefn2(o,N1,n); E=seqreverse(Jm);
A(i,1:N)=[Jm zeros(1,N-N1)];
A(N-i+1,1:N)=[zeros(1,N-N1) E];end
A(o/2,1:N)=Jm1(2:NN-1);end
% The matrix J1 to the power  O/2 ; rec=A;
```

The Modified Second-Order Finite-Difference Method 5.1:

```
% To Approximate The Solution for Non-linear Even-Order BVP.
% With Boundary Conditions d(2i)W0 ,d(2i)WN+1.
```

```
% INPUTS:- The Function For BVP f(X,W)
%         - The Derivative of f(X,W) with respect to W
%         - The Interval End-points
%         - The Number of Subinterval
%         - The Problem Order
%         - The Boundary Condition for problem
%         - The Maximum Number of Iterations
%         - The Tolerance
% OUTPUTS:- The Approximated Solution For Problem.
%          - The Associated Error For Approximated Wn
syms('X','W');
F=input('Write The F(xn,wn)for Problem:');
F2=input('Write the derivative for F(xn,wn) with respect to Wn:');
a=input('Write the initial point of interval a =');
b=input('Write the end point of interval b =');
N=input('Write the number of sub-interval[a,b] =');
o=input('Write The Problem Order =');
A0=input('Write The boundary condition as vector[W0 d2W0 d4W0 d6W0
d8W0]=');
```

```
B0=input('Write The boundary condition as vector[WN+1 d2WN+1
d4WN+1 d6WN+1 d8WN+1]=');

M=input('Write The Maximum Number of iteration=') ;

F3=input('Write The Exact Function for The Problem=');

T=input('Write The Tolerance You want=');

% Calculate the step-size

h=(b-a)/(N+1);

% The Initial Approximation for Wi

for i=1:N

   x(i)=a+i*h;

   w(i)=A0(1)+i*h*((A0(1)-B0(1))/(b-a));end

k=1

while (k<=M)

% The constants vector for problem using FN1 function for any even-order
BVPs

d=FN1(o,N,h,A0,B0);

% Evaluate The function and its derivative at W

Fn1=FmethodII(x,w,F);

Fn2=F2methodII(x,w,F2);
```

```
Fn3=ExactFn(x,F3);

%The Appropriate Matrix for system using function FN4 instead of J1^O/2

J1=FN4(o,N);

% The Jacobian Matrix for system

J=J1-diag(h^o*Fn2);

% The system function at Wi

f=-(J1*w'-h^o*Fn1'-d');

% The solution vector for system Jv=-f

v=linsolve(J,f);    t=norm(v);

if t<=T       break, end

% The New Approximated solution for system

w=w+v';  ww(k,1:N)=w;

k=k+1;

% The Error For approximated solution

e=abs(Fn3-w); end

% The approximated solution for the given BVP

fprintf(1,' X      WW                Error'), [x'  ww' e']
```

Program (5.4):

This program used to determine the entries of J_1 power with size N×N

% INPUTS: - The order of the problem.

% - The Matrix Size

%OUTPUTS: - The $J_1^{o/2}$ with size N×N

o=input('The Order of Differential Equation =');

N=input('The Number of Subinterval =');

N2=1+o/2 ,NN=o+1;

% This function used to find central finite-difference method

Jm1=coefn(o,NN);

if N>o-1

for i=1:o/2

n=I, N1=i+o/2; aN1=[-i+1:o/2]; aN2=-i*ones(1,N2);

% The programming function to determine the O/2 rows for $J_1^{o/2}$.

Jm=coefn2(o,N1,n,N);

E=seqreverse(Jm); A(i,1:N)=[Jm zeros(1,N-N1)];

A(N-i+1,1:N)=[zeros(1,N-N1) E]; end

for i=N2:N-o/2

```
A(i,1:N)=[zeros(1,i-N2) Jm1 zeros(1,(N-NN-i+N2))]; end

elseif N==o-1

for i=1:o/2-1

n=i;  N1=i+o/2;  aN1=[-i+1:o/2];  aN2=-i*ones(1,N2);

Jm=coefn2(o,N1,n,N);  E=seqreverse(Jm);

A(i,1:N)=[Jm zeros(1,N-N1)]; A(N-i+1,1:N)=[zeros(1,N-N1) E];end

A(o/2,1:N)=Jm1(2:NN-1);end

disp('The O/2 Power of J1 Is:');A

% The Programming Function To Determine O/2 Rows of J1 To The
Power O/2

function [conf2]=coefn2(o,N1,n,N)

NN=o+1;

% The Central Finite-Difference Formula

V=coefn(o,NN);

if N>o-1 & n==o/2      VV1=V(2:NN);

elseif N==o-1 & n==o/2     VV1=V(2:NN-1);

else    II=(NN-(o/2)+n+1); II1=(N1-(o/2)+n);   II2=(NN-n-(o/2)+1);

v1=V(II:NN);  v2=[v1 zeros(1,II1)]; VV1=V(II2:NN)-v2;  end

conf2=round(VV1);
```

% This Function Is Used To Determine Central Finite-Difference Approximation

cofn1-Programming Function:

```
function [cofn1]=coefn(o,NN)

format rat

NN=o+1,fo=factorial(o),

%aNN= The h Coefficients For W(x+ah) Vector For Central Formula

%bNN=The Constant Vector To The System

aNN(1)=-o/2;

for i=2:NN

aNN(i)=aNN(1)+(i-1);end

for i=1:NN

if i==o+1   bNN(i)=fo;

else bNN(i)=0; end, end

% The Matrix of the h coefficients powers

for i=1:NN

for j=1:NN

if i==1      aaNN(i,j)=1;

else       aaNN(i,j)=aNN(j)^(i-1);  end, end, end
```

```
%The Coefficient Vector For W(x+ah)

cNN=linsolve(aaNN,bNN'); cofn1=(cNN');
```

Appendix (I)

This appendix contains the computer programs for chapter one

Program(1.1).m:

```
%Deriving Finite-Difference Approximations.

%Using undetermined coefficients.

% INPUTS: - The Order of Derivative.

%         - The Number & Coefficient of h for Y(x+ah).

% OUTPUTS:- The Coefficients for Y(x+ah).

%         - The Local Truncation Error

format rat

o=input('The Order of Differential Equation =');

n=input('Number of terms =');

if n<o+1

display('The number of terms must be greater than the order of derivative');

  n=input('Number of terms =');end

a=input('The coefficient of h for Y(x+ah) as vector between [  ]  =');

No=size(a);

if No<n |No>n
```

```
display('You miss some terms please input all terms you want');

a=input('The coefficient of h for Y(x+ah) as vector between [  ]  =');

h=sym('h','unreal');c=1/(h^o); fo=factorial(o);

%The constant vector for the system

for i=1:n

if i==o+1     b(i)=fo;

else b(i)=0;    end, end

% The coefficients of h matrix of Y(x+ah)

for i=1:n+o+2

   for j=1:n

      if i==1    aa(i,j)=1;

       else     aa(i,j)=a(j)^(i-1);        end,     end, end

% The coefficients of Y(x+ah)

co1=linsolve(aa(1:n,1:n),b');

% process to release the error

[x,y]=rat(co1);

co=round(x)./round(y);

% process to deal with errors result from machine and the using of format
% rat to find the local truncation error for formula
```

```
t1=0;  k=o+2;
while t1==0
tn=aa(k,1:n)*co/factorial(k-1);
if tn==0   k=k+1;
else   t1=lte1(co,n,aa,k);
if t1~=0 break   end
k=k+1; end, end
% The local truncation error of the finite-difference formula
tn=aa(k,1:n)*co/factorial(k-1);
% The order of the local truncation error
c1=h^(k-o-1);
fprintf('The Coefficients of u(x+ah) as you arrange are respectively\n');co'
disp('Multiplying By:');c
display('With Local Truncation Error');tn
disp('Multiplying By:');c1
```

Program(1.2).m:

This program is used for deriving finite-difference approximations for derivative using undetermined coefficients method with respect to Yn and applying derivatives.

```
 %INPUTS: - The Order Of Derivative.

%The Number & Coefficients of h for Y(x+ah)

%The Number & Derivative & Coefficients of h for Dy(x+ah)

 %OUTPUTS: - The Coefficients For Y(x+ah)

 -        %The Coefficients For DY(x+ah)

fprintf(1,'This is the Finite-Difference Method Approximations for
Derivative.\n')

fprintf(1,'Using undetermined coefficients.\n')

fprintf(1,'with respect to Yn and applying any derivatives. \n');

format rat

o=input('The Order of Derivative =');

n1=input('Number of Yn terms =');

an=input('The coefficients of h for y(x+ah)as vector between [  ]  =');

nn=size(an);

if nn<n1 | nn>n1
```

```
display('You write the terms wrong. Please input all terms you want exactly');

an=input('The coefficient of h for Y(x+ah) as vector between [  ]  =');

end

m=input('Number of derivatives =');

if m+n1<o+1

display('The number of Y terms and of derivatives must be greater than Order of the problem');

break, end

if m >0

v=input('The derivatives you want=');

am=input('The coefficient of h for Dy(x+ah)as vector between [  ]  =');

h=sym('h','unreal'); c=1/(h^o);

fo=factorial(o); mn=m+n1;

 %The constant vector for system

for i=1:mn

   if i==(o+1)          b(i)=fo;

   else b(i)=0;    end, end

 %The coefficients of h matrix for Y(x+ah)
```

```
for i=1:mn+2*o
   for j=1:n1
      if (i==1)&(j<=n1)        aa(i,j)=1;
      elseif j<=n1                  aa(i,j)=an(j)^(i-1);   end , end, end
 %The coefficients of h matrix for diY(x+ah)
for i=1:mn+2*o
   for j=1:m
      if i<v(j)+1            aam(i,j)=0؛
   elseif  i==v(j)+1       aam(i,j)=(factorial(v(j)));
  elseif j<i aam(i,j)=am(j)^(i-v(j)-1)*(factorial(i-1)/factorial(abs(i-v(j)-1)));
   end , end ,end
 % process to deal with some cases when the No. derivative=1 and the
derivative greater than the order of derivatives
if (m==1 & v(m)>o) |(m+n1)<v(m)    An=(1:n1); Am=v+ones(1,m);
for i=1:mn
   if i<n1+1   F(i,1:n1)=aa(An(i),1:n1);F1(i,1:m)=aam(An(i),1:m);
   elseif (i>n1)&(i<=mn) F(i,1:n1)=aa(Am(i-n1),1:n1);
      F1(i,1:m)=aam(Am(i-n1),1:m);    end, end
 %The matrix of h coefficients for Y(x+ah) & diY(x+ah)
```

```
A=[F F1 b];
 %To test if the matrix singular or not.
Ron1=det([F F1]);
if Ron1==0    break ;display(' repeat input');end
else A=[aa(1:mn,1:n1) aam(1:mn,1:m) b'];end
 %The coefficients of Y(x+ah) and derivatives we want
co1=linsolve(A(1:mn,1:mn),b');
for i=1:m
dc(i)=h^((v(i)));end
 %To prevent machine error & format rat error
[x,y]=rat(co1);   co=round(x)./round(y);t1=0;k=o+2
while t1==0
A1=[aa(k,1:n1) aam(k,1:m)];    tn=A1*co/factorial(k-1)
if tn==0 k=k+1;
else   n=mn;    A1;    t1=lte2(co,n,A1,k)
if t1~=0 break ;end
k=k+1;end;end
 %The local truncation error of finite-difference formula
```

```
tn=A1*co/factorial(k-1)

 %The order of the local truncation error

c1=h^(k-o-1);

fprintf('The coefficients for Yn are respectively = \n');cof=co(1:n1');

fprintf('Multiplying By: \n');c

fprintf('The derivatives coefficients are respectively = \n');co(n1+1:mn')

fprintf('Multiplying by :\n');dc

fprintf('The Associated Local Truncation Error is = \n'); tn

fprintf('Multiplying by :\n');c1

elseif m==0

fprintf('Please This Program cannot do this process. Use Algorithm(1.2)\n')

else fprintf('Sorry The Number of Derivatives is False !!!!!!!\n');end
```

Program (1.3).m:

We enter this algorithm through the programs to find the finite –difference approximation which gets the error and it is order.

```
% The steps to determine the associated local truncation error

% aa : is the coefficient h matrix

% co :is the coefficient of W(x+ih) the program.

t1=0;k=o+2;
```

```
%This loop to  find the first coefficient (tn) not equal to zero

while t1==0

tn=aa(k,1:n)*co/factorial(k-1);

if tn==0 k=k+1;

else

%ltel : is a programming function to test if the calculated coefficient for
%LTE=0 or not

t1=lte1(co,n,aa,k);

% This if statement to break loop if the value of tn is not zero

if t1~=0 break ,end

k=k+1;end,end

% The LTE for the formula

tn=aa(k,1:n)*co/factorial(k-1);

% The Order of the error

c1=h^(k-o-1);

display('With Local Truncation Error');tn

disp('Multiplying By:');c1

function [rec]=lte1(co,n,aa,k)

% discrete fraction to nominator and denominator
```

```
[e1,e2]=rat(co);e11=round(e1);e22=round(e2);
% Steps to get the least common divisor for denominator vector
ne2=length(e22);k1=1;me2=max(e22);lme=e22;te=1;
while te~=0
for i=1:ne2
lme(i)=lcm(me2,lme(i));end
me2=max(lme);   g=me2*ones(1,ne2);    te=max(abs(lme-g'));
   if te==0        break,end
 k1=k1+1;end
% The new vector nominator after we get the same denominator
ce2=round(lme./e22);co2=e11.*ce2;
% The error for the nominator vector
tt=aa(k,1:n)*co2;rec=tt;
```

Appendix (II)

This appendix contains the computer programs for chapter two.

Program (2.1).m:

```
LINEAR FINITE-DIFFERENCE METHOD (2.1)

 To approximate the solution for the linear boundary-value problem

Y''' = F(X,Y,Y',Y''), a<=X<=b, Y(a) = ALPHA0, Y'(a) =ALPHA1

Y(b)=BETA0

% INPUT:   Endpoints A,B; boundary conditions a0,a1,b0.

% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1

format

% Step(1):determines step-size h and controls it

for k= 1:4

N(k)=(10*2^(k-1))-1;a=0;b=1;N1=N(1);h=(b-a)/(N(k)+1)

% The boundary conditions value

a0=1;b0=0;a1=1;

%The associated functions with derivative in problem and the remainder

for i=1:N(k)

x(i)=a+i*h;    p2(i)=(2*x(i)^2);    p1(i)=-(3*x(i));    p0(i)=(-5*x(i)^2);
```

```
r(i)=2*h^3*(exp(2*x(i))*(3*x(i)^3-x(i)^2-5*x(i)-4));end

%The finite-difference formula for 1st and 2nd derivative in problem

for i=1:N(k)

F1(i)=2*h*p2(i)-h^2*p1(i);    F2(i)=-4*h*p2(i)+2*h^3*p0(i);

F3(i)=2*h*p2(i)+h^2*p1(i);end

%The diagonals for matrix A for system Aw=c

d0(1)=-(18+3*F2(1));dU1(1)=-3*F3(1);dU2(1)=2;

dL1(1)=54-3*F1(2);dL2(1)=-6;dL3(1)=1;

for i=2:N(k)

if i==2    d0(i)=-(36+3*F2(i));    dU1(i)=10-3*F3(i);    dU2(i)=0;

dL1(i)=12-F1(i+1);    dL2(i)=-6;    dL3(i)=1;

elseif i>2 & i<=N(k)-3

d0(i)=-(10+F2(i));    dU1(i)=3-F3(i);    dU2(i)=0;    dL1(i)=12-F1(i+1);

dL2(i)=-6;    dL3(i)=1;

elseif i==N(k)-2    d0(i)=-(10+F2(i));    dU1(i)=3-F3(i);    dU2(i)=0;
dL1(i)=12-F1(i+1);    dL2(i)=-6;

 elseif i==N(k)-1   d0(i)=-(10+F2(i)); dU1(i)=3-F3(i);    dL1(i)=12-F1(i+1);

elseif i==N(k)     d0(i)=-(10+F2(i));    end, end

% The constant vector c
```

```
c1=3*r(1)+(-16+3*F1(1))*a0-12*h*a1;c2=3*r(2)+28*a0+12*h*a1;

c3=r(3)-a0;cN=r(N(k))+(-3+F3(N(k)))*b0;

C=[c1 c2 c3 r(4:(N(k)-1)) cN ];

% Factorize The Matrix A=LU

U2=dU2;L0=ones(1,N(k));U1(1)=dU1(1);

U0(1)=d0(1);L1(1)=dL1(1)/U0(1);L2(1)=dL2(1)/U0(1);L3=dL3(1)/U0(1);

for i=2:N(k)

if i==2    U0(i)=d0(i)-U1(i-1)*L1(i-1);   U1(i)=dU1(i)-U2(i-1)*L1(i-1);

L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);  L3(i)=dL3(i)/U0(i);

elseif  i>2 &  i<=N(k)-3   U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2);

U1(i)=dU1(i)-U2(i-1)*L1(i-1);

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i); L3(i)=dL3(i)/U0(i);

elseif i==N(k)-2    U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2);

U1(i)=dU1(i)-U2(i-1)*L1(i-1);

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);
```

```
elseif i==N(k)-1
U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2);
U1(i)=dU1(i)-U2(i-1)*L1(i-1);
L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
elseif i==N(k)   U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2); end,end
%To solve AW=dd using factorized matrix
%Step(1):We solve LY1=dd;
y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);
y1(3)=C(3)-(L2(1)*y1(1)+L1(2)*y1(2));
for i=4:N(k)
y1(i)=C(i)-(L3(i-3)*y1(i-3)+L2(i-2)*y1(i-2)+L1(i-1)*y1(i-1));end
%Step(2):We solve UW=Y1
ww(N(k))=y1(N(k))/U0(N(k));
ww(N(k)-1)=(y1(N(k)-1)-U1(N(k)-1)*ww(N(k)))/U0(N(k)-1);
for i=2:N(k)-1
ww(N(k)-i)=(y1(N(k)-i)-U1(N(k)-i)*ww(N(k)-i+1)-U2(N(k)-i)*ww(N(k)-
i+2))/U0(N(k)-i);end
W1(1:N1,k)=ww(2^(k-1):2^(k-1):N(k));end
% The Exact solution for y and X
```

```
for i=1:9

   h=(b-a)/10;    X(i)=a+i*h;    Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;

Ext22=(4*W1(1:N1,3)-W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;

Ext31=(16*Ext22-Ext21)/15;  Ext32=(16*Ext23-Ext22)/15;

Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext21'; Ext31']

plot(X,W1(1:N1,4),'red',X,Ext31,'blue',X,Y,'green')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f
%12.8f\n',MR );
```

Program (2.2).m:

```
LINEAR FINITE-DIFFERENCE METHOD (2.2)

% To approximate the solution for the linear boundary-value problem

% Y''' = F(X,Y,Y',Y''), a<=X<=b, Y(a) = ALPHA0, Y(b)=BETA0, Y''(b)
=BETA2

% INPUT:   Endpoints A,B; boundary conditions a0,a1,b0.
```

```
% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1
format
% Step(1):determines step-size h and controls it
for k= 1:4
N(k)=(10*2^(k-1))-1;a=0;b=1;N1=N(1);h=(b-a)/(N(k)+1)
% The boundary conditions value
a0=1;b0=0;b2=-4*exp(2);
%The associated functions with derivative in problem and the remainder
for i=1:N(k)
x(i)=a+i*h; p2(i)=(2*x(i)^2); p1(i)=-(3*x(i));  p0(i)=-(5*x(i)^2);
r(i)=2*h^3*(exp(2*x(i))*(3*x(i)^3-x(i)^2-5*x(i)-4));end
%The finite-difference formula for 1st and 2nd derivative in problem
for i=1:N(k)
F1(i)=2*h*p2(i)-h^2*p1(i); F2(i)=-4*h*p2(i)+2*h^3*p0(i);
F3(i)=2*h*p2(i)+h^2*p1(i);end
%The diagonals for matrix A for system Aw=c
d0(1)=10-F2(1);dU1(1)=-(12+F3(1));dU2(1)=6;dU3(1)=-1;dL1(1)=-
(3+F1(2));dL2(1)=0;
for i=2:N(k)
```

```
if i>1 & i<=(N(k)-3)   d0(i)=10-F2(i);  dU1(i)=-(12+F3(i));  dU2(i)=6;
dU3(i)=-1; dL1(i)=-(3+F1(i+1));    dL2(i)=0;
elseif i==(N(k)-2)    d0(i)=10-F2(i);    dU1(i)=-(12+F3(i));    dU2(i)=6;
dL1(i)=-(34+11*F1(i+1));   dL2(i)=-10;
elseif i==N(k)-1  d0(i)=(114-11*F2(i));  dU1(i)=-(126+11*F3(i));
dL1(i)=18-11*F1(i+1);
elseif i==N(k)    d0(i)=-(6+11*F2(i));    end,end
% The constant vector c
c1=r(1)+(3+F1(1))*a0;cN2=r(N(k)-2)+b0;
cN1=11*r(N(k)-1)-46*b0+12*h^2*b2;
cN=11*r(N(k))+(2+11*F3(N(k)))*b0-12*h^2*b2;
C=[c1 r(2:(N(k)-3)) cN2 cN1 cN ];
% Factorize The Matrix A=LU
U3=dU3;L0=ones(1,N(k));
U1(1)=dU1(1);U0(1)=d0(1);L1(1)=dL1(1)/U0(1);L2(1)=dL2(1)/U0(1);
U2(1)=dU2(1);
for i=2:N(k)
if i==2   U0(i)=d0(i)-(L1(i-1)*U1(i-1)); U1(i)=dU1(i)-(L1(i-1)*U2(i-1));
```

```
U2(i)=dU2(i)-(L1(i-1)*U3(i-1));  L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=dL2(i)/U0(i);

elseif i>2 &i<N(k)-2

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)); L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=dL2(i)/U0(i);

elseif i==N(k)-2     U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)); L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=dL2(i)/U0(i);

elseif i==N(k)-1     U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

elseif i==N(k)    U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)); end, end

%To solve AW=dd using factorized matrix

%Step(1):We solve LY1=dd;

y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);
```

```
for i=3:N(k)

y1(i)=C(i)-(L2(i-2)*y1(i-2)+L1(i-1)*y1(i-1));end

%Step(2):We solve UW=Y1

w(N(k))=y1(N(k))/U0(N(k));w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-
1)*w(N(k)))/U0(N(k)-1);

w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);

w(N(k)-3)=(y1(N(k)-3)-(U1(N(k)-3)*w(N(k)-2)+U2(N(k)-3)*w(N(k)-
1)+U3(N(k)-3)*w(N(k))))/U0(N(k)-3);

w(N(k)-4)=(y1(N(k)-4)-(U1(N(k)-4)*w(N(k)-3)+U2(N(k)-4)*w(N(k)-
2)+U3(N(k)-4)*w(N(k)-1)))/U0(N(k)-4);

for i=5:N(k)-1

cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2)+U3(N(k)-
i)*w(N(k)-i+3);

w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end

W1(1:N1,k)=w(2^(k-1):2^(k-1):N(k));end

% The Exact solution for y and X

for i=1:9

h=(b-a)/10;    X(i)=a+i*h;    Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution
```

```
Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;

Ext22=(4*W1(1:N1,3)-W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;

Ext31=(16*Ext22-Ext21)/15;Ext32=(16*Ext23-Ext22)/15;

Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext31'; Ext41'];

plot(X,W1(1:N1,4),'red',X,Ext41,'blue',X,Y,'green')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f
%12.8f\n',MR );
```

Program (2.3).m:

```
LINEAR FINITE-DIFFERENCE METHOD(2.3)

% To approximate the solution for the linear boundary-value problem

%    Y''' = F(X,Y,Y',Y''), a<=X<=b, Y(a) = ALPHA0, Y'(a) =ALPHA1

%    Y''(b)=BETA2

% INPUT:   Endpoints A,B; boundary conditions a0,a1,b2.

% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1

format

% Step(1):determines step-size h and controls it
```

```
for k= 1:8

N(k)=(10*2^(k-1))-1;a=0;b=1;N1=N(1);h=(b-a)/(N(k)+1);

% Case(III) :The boundary conditions value

a0=1;a1=1;b2=-4*exp(2);

%The associated functions with derivative in problem and the remainder

for i=1:N(k)

x(i)=a+i*h; p2(i)=(2*x(i)^2); p1(i)=-(3*x(i)); p0(i)=-(5*x(i)^2);

r(i)=2*h^3*(exp(2*x(i))*(3*x(i)^3-x(i)^2-5*x(i)-4));end

%The finite-difference formula for 1st and 2nd derivative in problem

for i=1:N(k)-1

F1(i)=2*h*p2(i)-h^2*p1(i);  F2(i)=-4*h*p2(i)+2*h^3*p0(i);

 F3(i)=2*h*p2(i)+h^2*p1(i);end

G0=35*(h*p2(N(k))+h^2*p1(N(k)));G1=35*h*p2(N(k));

G2=-70*(h*p2(N(k))+h^2*p1(N(k)));

G3=35*(h*p2(N(k))+2*h^2*p1(N(k))+2*h^3*p0(N(k)));

%The diagonals for matrix A for system Aw=c

d0(1)=-(18+3*F2(1));dU1(1)=-3*F3(1);dU2(1)=2;dL1(1)=54-
3*F1(2);dL2(1)=-6;dL3(1)=1;

for i=2:N(k)
```

```
if i==2    d0(i)=-(36+3*F2(i));    dU1(i)=10-3*F3(i);    dU2(i)=0;

dL1(i)=12-F1(i+1);    dL2(i)=-6;    dL3(i)=1;

elseif i>2 & i<N(k)-3   d0(i)=-(10+F2(i));   dU1(i)=3-F3(i);   dU2(i)=0;

dL1(i)=12-F1(i+1);  dL2(i)=-6;    dL3(i)=1;

elseif i==N(k)-3   d0(i)=-(10+F2(i));  dU1(i)=3-F3(i);   dU2(i)=0;

dL1(i)=12-F1(i+1);  dL2(i)=-6;    dL3(i)=2;

elseif i==N(k)-2   d0(i)=-(10+F2(i)); dU1(i)=3-F3(i);   dU2(i)=0;

dL1(i)=12-F1(i+1); dL2(i)=-(42+G1);

elseif i==N(k)-1   d0(i)=-(10+F2(i));  dU1(i)=3-F3(i);  dL1(i)=78-G2;

elseif i==N(k)  d0(i)=-(38+G3);    end,end

% The constant vector c

c1=(-16+3*F1(1))*a0+3*r(1)-12*h*a1;c2=28*a0+3*r(2)+12*h*a1;

c3=r(3)-a0;cN=35*r(N(k))+(-36+G0)*b2*h^2;

C=[c1 c2 c3 r(4:N(k)-1) cN];

% Factorize The Matrix A=LU

U2=dU2;L0=ones(1,N(k));U1(1)=dU1(1);

U0(1)=d0(1);L1(1)=dL1(1)/U0(1);L2(1)=dL2(1)/U0(1);L3=dL3(1)/U0(1);

for i=2:N(k)
```

```
if i==2   U0(i)=d0(i)-U1(i-1)*L1(i-1);  U1(i)=dU1(i)-U2(i-1)*L1(i-1);
L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);
L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);  L3(i)=dL3(i)/U0(i);
elseif  i>2 &  i<=N(k)-3  U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2);
U1(i)=dU1(i)-U2(i-1)*L1(i-1);
L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);  L3(i)=dL3(i)/U0(i);
elseif i==N(k)-2    U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2);
U1(i)=dU1(i)-U2(i-1)*L1(i-1);
L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);
elseif i==N(k)-1       U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2);
U1(i)=dU1(i)-U2(i-1)*L1(i-1);
L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
elseif i==N(k)    U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2);end, end
%To solve AW=dd using factorized matrix
%Step(1):We solve LY1=dd;
y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);y1(3)=C(3)-L1(2)*y1(2)-L2(1)*y1(1);
```

```
for i=4:N(k)

y1(i)=C(i)-(L3(i-3)*y1(i-3)+L2(i-2)*y1(i-2)+L1(i-1)*y1(i-1));end

%Step(2):We solve UW=Y1

w(N(k))=y1(N(k))/U0(N(k));w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-
1)*w(N(k)))/U0(N(k)-1);

w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);

for i=3:N(k)-1

cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2);

w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end

W1(1:N1,k)=w(2^(k-1):2^(k-1):N(k));end

% The Exact solution for y and X

for i=1:9

h=(b-a)/10;    X(i)=a+i*h;    Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;

Ext22=(4*W1(1:N1,3)-W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;

Ext24=(4*W1(1:N1,5)-W1(1:N1,4))/3;
```

```
Ext25=(4*W1(1:N1,6)-W1(1:N1,5))/3;

Ext26=(4*W1(1:N1,7)-W1(1:N1,6))/3;

Ext27=(4*W1(1:N1,8)-W1(1:N1,7))/3;Ext31=(16*Ext22-Ext21)/15;
Ext32=(16*Ext23-Ext22)/15;Ext33=(16*Ext24-Ext23)/15;
Ext34=(16*Ext25-Ext24)/15;Ext35=(16*Ext26-Ext25)/15;

Ext41=(64*Ext32-Ext31)/63;   Ext42=(64*Ext33-Ext32)/63;
Ext43=(64*Ext34-Ext33)/63;   Ext44=(64*Ext35-Ext34)/63;

Ext51=(256*Ext42-Ext41)/255;Ext52=(256*Ext43-Ext42)/255;

Ext53=(256*Ext44-Ext43)/255;Ext61=(1024*Ext52-Ext51)/1023;
Ext62=(1024*Ext53-Ext52)/1023;Ext71=(4096*Ext62-Ext61)/4095;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,6)';W1(:,7)';W1(:,8)';Ext21';
Ext31';Ext41';Ext51';Ext61';Ext71'];

plot(X,W1(1:N1,8),'red',X,Ext71,'blue',X,Y,'green')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f\n',MR(1:6,:));

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f
%12.8f\n',[X;Y;MR(7:12,:)]);
```

Program (2.4).m:

```
LINEAR FINITE-DIFFERENCE METHOD (2.4)

% To approximate the solution for the linear boundary-value problem

%   Y''' = F(X,Y,Y',Y''), a<=X<=b, Y'(a) = ALPHA1, Y'(b) =ALPHA1

%   Y(b)=BETA0

% INPUT:   Endpoints A,B; boundary conditions a0,a1,b0.

 % OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1
% Step(1): determines step-size h and controls it

for k= 1:4

N(k)=(10*2^(k-1))-1;a=0;b=1;N1=N(1);h=(b-a)/(N(k)+1)

% The boundary conditions value

a1=1;b0=0;b1=-exp(2);

%The associated functions with derivative in problem and the remainder

for i=1:N(k)

 x(i)=a+i*h; p2(i)=(2*x(i)^2);    p1(i)=-(3*x(i));    p0(i)=(-5*x(i)^2);

r(i)=2*h^3*(exp(2*x(i))*(3*x(i)^3-x(i)^2-5*x(i)-4));end

%The finite-difference formula for 1st and 2nd derivative in problem

for i=2:N(k)

F1(i)=2*h*p2(i)-h^2*p1(i); F2(i)=-4*h*p2(i)+2*h^3*p0(i);
```

```
F3(i)=2*h*p2(i)+h^2*p1(i);end

G0=-450*h*p2(1)+275*h^2*p1(1);

G1=-(300*h*p2(1)+550*h^2*p1(1)-825*h^3*p0(1));

G2=150*h*p2(1)+550*h^2*p1(1);G3=150*h*p2(1);

%The diagonals for matrix A for system Aw=c

d0(1)=1749-G1;dU1(1)=-(3168+G2);dU2(1)=(1683-G3);dU3(1)=-264;
dL1(1)=-3-F1(2);dL2(1)=0;

for i=2:N(k)

if i>1 & i<=N(k)-3   d0(i)=10-F2(i); dU1(i)=-12-F3(i); dU2(i)=6;
dU3(i)=-1;  dL1(i)=-3-F1(i+1);    dL2(i)=0;

elseif i==N(k)-2   d0(i)=10-F2(i); dU1(i)=-12-F3(i); dU2(i)=6;

dL1(i)=-10-3*F1(i+1); dL2(i)=-2;

elseif i==N(k)-1 d0(i)=36-3*F2(i); dU1(i)=-54-3*F3(i);dL1(i)=-3*F1(i+1);

elseif i==N(k)   d0(i)=18-3*F2(i);  end,end

% The constant vector c

c1=(825/2)*r(1)+(-594+G0)*h*a1;cN2=b0+r(N(k)-2);

cN1=-28*b0+12*h*b1+3*r(N(k)-1);

cN=3*r(N(k))+(16+3*F3(N(k)))*b0-12*h*b1;

C=[c1 r(2:(N(k)-3)) cN2 cN1 cN ];
```

```
% Factorize The Matrix A=LU

U3=dU3;L0=ones(1,N(k));U1(1)=dU1(1);U0(1)=d0(1);L1(1)=dL1(1)/U0(1
);L2(1)=dL2(1)/U0(1);U2(1)=dU2(1);

for i=2:N(k)

if i==2  U0(i)=d0(i)-(L1(i-1)*U1(i-1));  U1(i)=dU1(i)-(L1(i-1)*U2(i-1));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)); L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

 L2(i)=dL2(i)/U0(i);

elseif i>2 &i<N(k)-2     U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1));L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=dL2(i)/U0(i);

elseif i==N(k)-2   U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1));L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=dL2(i)/U0(i);

elseif i==N(k)-1    U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

elseif i==N(k)  U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));    end, end
```

```
%To solve AW=dd using factorized matrix
%Step(1):We solve LY1=dd;
y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);
for i=3:N(k)   y1(i)=C(i)-(L2(i-2)*y1(i-2)+L1(i-1)*y1(i-1));end
%Step(2):We solve UW=Y1
w(N(k))=y1(N(k))/U0(N(k));w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-
1)*w(N(k)))/U0(N(k)-1);
w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);
w(N(k)-3)=(y1(N(k)-3)-(U1(N(k)-3)*w(N(k)-2)+U2(N(k)-3)*w(N(k)-
1)+U3(N(k)-3)*w(N(k))))/U0(N(k)-3);
w(N(k)-4)=(y1(N(k)-4)-(U1(N(k)-4)*w(N(k)-3)+U2(N(k)-4)*w(N(k)-
2)+U3(N(k)-4)*w(N(k)-1)))/U0(N(k)-4);
for i=5:N(k)-1
cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2)+U3(N(k)-
i)*w(N(k)-i+3);
w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end
W1(1:N1,k)=w(2^(k-1):2^(k-1):N(k));end
% The Exact solution for y and X
for i=1:9
```

```
   h=(b-a)/10;    X(i)=a+i*h;   Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;

Ext22=(4*W1(1:N1,3)-W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;Ext31=(16*Ext22-Ext21)/15;
Ext32=(16*Ext23-Ext22)/15;Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext31'; Ext41']

plot(X,W1(1:N1,4),'red',X,Ext41,'blue',X,Y,'green')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f
%12.8f\n',MR );
```

Appendix (III)

This appendix contains the computer programs for chapter three

Program (3.1).m:

```
LINEAR FINITE-DIFFERENCE METHOD (3.1)

% To approximate the solution for the linear boundary-value problem

%    Y"" = F(X,Y,Y',Y"'), a<=X<=b, Y(a) = ALPHA0, Y(b) = BETA0

%    Y"(a)=ALPHA2 ,Y"(b)= BETA2

% INPUT:   Endpoints A,B; boundary conditions a0,a2,b0,b2.

% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1

format

% Step(1): determines steps-size h and controls it

for k= 1:4

N(k)=(10*2^(k-1))-1;a=0;b=1;N1=N(1);h=(b-a)/(N(k)+1)

% The boundary conditions value

a0=1;a2=0;b0=0;b1=-exp(2);

%The associated functions with derivative in problem and the remainder

for i=1:N(k)

x(i)=a+i*h;    p3(i)=h*(1+2*x(i)^2); p2(i)=h^2*(-3*x(i));
```

```
p1(i)=h^3*(-5*x(i)^2);    p0(i)=h^4*(1+x(i)^3);

r(i)=h^4*(exp(2*x(i))*(x(i)^4+5*x(i)^3+x(i)^2-7*x(i)-13));end

%The finite-difference formula for 1st and 2nd derivative in problem

for i=2:N(k)-1

F1(i)=-p3(i);  F2(i)=2*p3(i)+2*p2(i)-p1(i); F3(i)=-4*p2(i)+2*p0(i);

F4(i)=-2*p3(i)+2*p2(i)+p1(i);end

G0=10*p3(1)+110*p2(1)-55*p1(1);G1=30*p3(1)-220*p2(1)+110*p0(1);

G2=-90*p3(1)+110*p2(1)+55*p1(1);G3=50*p3(1);G02=-60*p3(1);

GN0=-32*p3(N(k))+12*p2(N(k))+6*p1(N(k));

GN3=36*p3(N(k))-24*p2(N(k))+12*p0(N(k));

GN2=12*p2(N(k))-6*p1(N(k));GN1=-4*p3(N(k));GN01=24*p3(N(k));

%The diagonals for matrix A for system Aw=c

d0(1)=704-G1;dU1(1)=-(594+G2);dU2(1)=(176-G3);dU3(1)=-11;dL1(1)=-
8-F2(2);dL2(1)=2-F1(3);dL3(1)=0;

for i=2:N(k)

if i>1 & i<N(k)-3    d0(i)=12-F3(i);    dU1(i)=-8-F4(i);   dU2(i)=2+F1(i);

dU3(i)=0; dL1(i)=-8-F2(i+1); dL2(i)=2-F1(i+2); dL3(i)=0;

elseif i==N(k)-3  d0(i)=12-F3(i); dU1(i)=-8-F4(i);   dU2(i)=2+F1(i);

dU3(i)=0; dL1(i)=-8-F2(i+1);    dL2(i)=2-F1(i+2); dL3(i)=-3;
```

```
elseif i==N(k)-2  d0(i)=12-F3(i);    dU1(i)=-8-F4(i);    dU2(i)=2+F1(i);
dL1(i)=-8-F2(i+1);   dL2(i)=32-GN1;
elseif i==N(k)-1   d0(i)=12-F3(i);  dU1(i)=-8-F4(i);    dL1(i)=-108-GN2;
elseif i==N(k)  d0(i)=192-GN3;    end,end
% The constant vector c
c1=110*r(1)+(275+G0)*a0+(-132+G02)*h^2*a2;
c2=(-2+F1(2))*a0+2*r(2);cN1=(-2-F1(N(k)-1))*b0+2*r(N(k)-1);
cN=12*r(N(k))+(113+GN0)*b0+(-60+GN01)*h*b1;
C=[c1 c2 2*r(3:(N(k)-2)) cN1 cN ];
% Factorize The Matrix A=LU for 2 upper and 3 lower
U3=dU3;U2(1)=dU2(1);L0=ones(1,N(k));U1(1)=dU1(1);
U0(1)=d0(1);L1(1)=dL1(1)/U0(1);L2(1)=dL2(1)/U0(1);L3=dL3(1)/U0(1);
for i=2:N(k)
if i==2   U0(i)=d0(i)-(L1(i-1)*U1(i-1)); U1(i)=dU1(i)-(L1(i-1)*U2(i-1));
U2(i)=dU2(i)-(L1(i-1)*U3(i-1));L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);
L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i); L3(i)=dL3(i)/U0(i);
elseif  i>2 &  i<=N(k)-3 U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));
U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));
```

```
U2(i)=dU2(i)-(L1(i-1)*U3(i-1));
L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);   L3(i)=dL3(i)/U0(i);
elseif i==N(k)-2   U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));
U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));
U2(i)=dU2(i)-(L1(i-1)*U3(i-1));
L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);
elseif i==N(k)-1    U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));
U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));
L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
elseif i==N(k)  U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2);       end,end
%To solve AW=dd using factorized matrix
%Step(1):We solve LY1=dd;
y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);
y1(3)=C(3)-(L2(1)*y1(1)+L1(2)*y1(2));
for i=4:N(k)
y1(i)=C(i)-(L3(i-3)*y1(i-3)+L2(i-2)*y1(i-2)+L1(i-1)*y1(i-1));end
```

```
%Step(2):We solve UW=Y1

w(N(k))=y1(N(k))/U0(N(k));w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-
1)*w(N(k)))/U0(N(k)-1);

w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);

w(N(k)-3)=(y1(N(k)-3)-(U1(N(k)-3)*w(N(k)-2)+U2(N(k)-3)*w(N(k)-
1)+U3(N(k)-3)*w(N(k))))/U0(N(k)-3);

w(N(k)-4)=(y1(N(k)-4)-(U1(N(k)-4)*w(N(k)-3)+U2(N(k)-4)*w(N(k)-
2)+U3(N(k)-4)*w(N(k)-1)))/U0(N(k)-4);

for i=5:N(k)-1

cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2)+U3(N(k)-
i)*w(N(k)-i+3);

w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end

W1(1:N1,k)=w(2^(k-1):2^(k-1):N(k));end

% The Exact solution for y and X

for i=1:9

h=(b-a)/10;   X(i)=a+i*h;    Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;

Ext22=(4*W1(1:N1,3)-W1(1:N1,2))/3;
```

```
Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;Ext31=(16*Ext22-Ext21)/15;
Ext32=(16*Ext23-Ext22)/15;Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext31'; Ext41']

plot(X,W1(1:N1,1),'red',X,Ext41,'blue',X,Y,'green')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f
%12.8f\n',MR );

% The Error of results

for i=1:k

ER1(i)=max(abs(Y'-W1(:,i)));end

ERT1=[max(abs(Y'-Ext21)) max(abs(Y'-Ext31)) max(abs(Y'-Ext41)) ]
```

Program (3.2).m:

```
 LINEAR FINITE-DIFFERENCE METHOD (3.2)

% To approximate the solution for the linear boundary-value problem

%    Y'''' = F(X,Y,Y',Y''), a<=X<=b, Y(a) = ALPHA0, Y(b) = BETA0

%   Y'(a)=ALPHA1 ,Y'(b)= BETA1

% INPUT:   Endpoints A,B; boundary conditions a0,a2,b0,b1.

% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1

format
```

```
% Step(1):determines steps-size h and controls it
for k= 1:4
N(k)=(10*2^(k-1))-1;a=0;b=1;N1=N(1);h=(b-a)/(N(k)+1)
% The boundary conditions value
a0=1;a1=1;b0=0;b1=-exp(2);
%The associated functions with derivative in problem and the remainder
for i=1:N(k)
x(i)=a+i*h;    p3(i)=h*(1+2*x(i)^2);    p2(i)=h^2*(-3*x(i));
p1(i)=h^3*(-5*x(i)^2);    p0(i)=h^4*(1+x(i)^3);
r(i)=h^4*(exp(2*x(i))*(x(i)^4+5*x(i)^3+x(i)^2-7*x(i)-13));end
%The finite-difference formula for 1st and 2nd derivative in problem
for i=2:N(k)-1
F1(i)=-p3(i);    F2(i)=2*p3(i)+2*p2(i)-p1(i);    F3(i)=-4*p2(i)+2*p0(i);
F4(i)=-2*p3(i)+2*p2(i)+p1(i);end
G0=32*p3(1)+12*p2(1)-6*p1(1);G1=-36*p3(1)-24*p2(1)+12*p0(1);
G2=12*p2(1)+6*p1(1);G3=4*p3(1);G01=24*p3(1);
GN0=-32*p3(N(k))+12*p2(N(k))+6*p1(N(k));
GN3=36*p3(N(k))-24*p2(N(k))+12*p0(N(k));
```

```
GN2=12*p2(N(k))-6*p1(N(k));GN1=-4*p3(N(k));GN01=24*p3(N(k));
%The diagonals for matrix A for system Aw=c
d0(1)=192-G1;dU1(1)=(-108-G2);dU2(1)=(32-G3);dU3(1)=-3;
dL1(1)=-8-F2(2);dL2(1)=2-F1(3);dL3(1)=0;
for i=2:N(k)
if i>1 & i<N(k)-3  d0(i)=12-F3(i);   dU1(i)=-8-F4(i);   dU2(i)=2+F1(i);
dU3(i)=0;   dL1(i)=-8-F2(i+1);   dL2(i)=2-F1(i+2);   dL3(i)=0;
elseif i==N(k)-3  d0(i)=12-F3(i);  dU1(i)=-8-F4(i); dU2(i)=2+F1(i);
dU3(i)=0;  dL1(i)=-8-F2(i+1);  dL2(i)=2-F1(i+2);   dL3(i)=-3;
elseif i==N(k)-2   d0(i)=12-F3(i);   dU1(i)=-8-F4(i);   dU2(i)=2+F1(i);
dL1(i)=-8-F2(i+1);  dL2(i)=32-GN1;
elseif i==N(k)-1  d0(i)=12-F3(i);   dU1(i)=-8-F4(i);   dL1(i)=-108-GN2;
elseif i==N(k)   d0(i)=192-GN3;   end,end
% The constant vector c
c1=12*r(1)+(113+G0)*a0+(60+G01)*h*a1;c2=(-2+F1(2))*a0+2*r(2);
cN1=(-2-F1(N(k)-1))*b0+2*r(N(k)-1);
cN=12*r(N(k))+(113+GN0)*b0+(-60+GN01)*h*b1;
C=[c1 c2 2*r(3:(N(k)-2)) cN1 cN ];
```

```
% Factorize The Matrix A=LU for 2 upper and 3 lower

U3=dU3;U2(1)=dU2(1);L0=ones(1,N(k));U1(1)=dU1(1);

U0(1)=d0(1);L1(1)=dL1(1)/U0(1);L2(1)=dL2(1)/U0(1);L3=dL3(1)/U0(1);

for i=2:N(k)

if i==2    U0(i)=d0(i)-(L1(i-1)*U1(i-1));U1(i)=dU1(i)-(L1(i-1)*U2(i-1));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1));L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);   L3(i)=dL3(i)/U0(i);

elseif  i>2 &  i<=N(k)-3 U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));U2(i)=dU2(i)-(L1(i-
1)*U3(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);  L3(i)=dL3(i)/U0(i);

elseif i==N(k)-2   U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));U2(i)=dU2(i)-(L1(i-
1)*U3(i-1)); L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);

elseif i==N(k)-1 U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
```

```
elseif i==N(k)   U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2);    end, end

%To solve AW=dd using factorization of matrix

%Step(1):We solve LY1=dd;

y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);

y1(3)=C(3)-(L2(1)*y1(1)+L1(2)*y1(2));

for i=4:N(k)

y1(i)=C(i)-(L3(i-3)*y1(i-3)+L2(i-2)*y1(i-2)+L1(i-1)*y1(i-1));end

%Step(2):We solve UW=Y1

w(N(k))=y1(N(k))/U0(N(k));w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-
1)*w(N(k)))/U0(N(k)-1);

w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);

w(N(k)-3)=(y1(N(k)-3)-(U1(N(k)-3)*w(N(k)-2)+U2(N(k)-3)*w(N(k)-
1)+U3(N(k)-3)*w(N(k))))/U0(N(k)-3);

w(N(k)-4)=(y1(N(k)-4)-(U1(N(k)-4)*w(N(k)-3)+U2(N(k)-4)*w(N(k)-
2)+U3(N(k)-4)*w(N(k)-1)))/U0(N(k)-4);

for i=5:N(k)-1

cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2)+U3(N(k)-
i)*w(N(k)-i+3);

w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end
```

```
W1(1:N1,k)=w(2^(k-1):2^(k-1):N(k));end

% The Exact solution for y and X

for i=1:9

 h=(b-a)/10;    X(i)=a+i*h;    Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;

Ext22=(4*W1(1:N1,3)-W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;Ext31=(16*Ext22-Ext21)/15;
Ext32=(16*Ext23-Ext22)/15;Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext31'; Ext41']

plot(X,W1(1:N1,1),'red',X,Ext41,'blue',X,Y,'green')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f\n',MR );
```

Program (3.3).m:

```
LINEAR FINITE-DIFFERENCE METHOD (3.3)

% To approximate the solution for the linear boundary-value problem

%    Y'''' = F(X,Y,Y',Y'''), a<=X<=b, Y(a) = ALPHA0, Y'(a) = ALPHA1

%    Y''(b)=BETA2 ,Y'''(b)= BETA3

% INPUT:   Endpoints A,B; boundary conditions a0,a1,b2,b3.
```

```
% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1
format
% Step(1):determines steps-size h and controls it
for k= 1:6
N(k)=(10*2^(k-1))-1; a=0 ;b=1 ;N1=N(1); h=(b-a)/(N(k)+1);
% The boundary conditions value
a0=1;a1=1;b2=-4*exp(2);b3=-12*exp(2);
%The associated functions with derivative in problem and the remainder
for i=1:N(k)
x(i)=a+i*h;   p3(i)=h*(1+2*x(i)^2);  p2(i)=h^2*(-3*x(i));
p1(i)=h^3*(-5*x(i)^2);   p0(i)=h^4*(1+x(i)^3);
r(i)=h^4*(exp(2*x(i))*(x(i)^4+5*x(i)^3+x(i)^2-7*x(i)-13)); end
%The finite-difference formula for 1st and 2nd derivative in problem
for i=2:N(k)-2
F1(i)=-p3(i);   F2(i)=2*p3(i)+2*p2(i)-p1(i);  F3(i)=-4*p2(i)+2*p0(i);
F4(i)=-2*p3(i)+2*p2(i)+p1(i);end
G0=32*p3(1)+12*p2(1)-6*p1(1);   G1=-36*p3(1)-24*p2(1)+12*p0(1);
G2=12*p2(1)+6*p1(1);   G3=4*p3(1);G01=24*p3(1);
```

```
GN14=-32*p3(N(k)-1)+290*p2(N(k)-1)+(145)*p1(N(k)-1);

GN13=-48*p3(N(k)-1)-580*p2(N(k)-1)+290*p0(N(k)-1);

GN12=192*p3(N(k)-1)+290*p2(N(k)-1)-(145)*p1(N(k)-1);

GN11=-112*p3(N(k)-1);GN102=144*p3(N(k)-1);GN103=-110*p3(N(k)-1);

GN02=2880*p3(N(k))+3306*p2(N(k))+2175*p1(N(k));

GN4=-3540*p3(N(k))+1044*p2(N(k))+4350*p1(N(k))+4350*p0(N(k));

GN3=7740*p3(N(k))-2088*p2(N(k))-4350*p1(N(k));

GN2=-4860*p3(N(k))+1044*p2(N(k));GN1=660*p3(N(k));

GN03=-750*p3(N(k))-2262*p2(N(k))-2900*p1(N(k));

%The diagonals for matrix A for system Aw=c

d0(1)=192-G1;dU1(1)=(-108-G2);dU2(1)=(32-G3);dU3(1)=-3;

dL1(1)=-8-F2(2);dL2(1)=2-F1(3);dL3(1)=0;

for i=2:N(k)

if i>1 & i<N(k)-3  d0(i)=12-F3(i);    dU1(i)=-8-F4(i);    dU2(i)=2+F1(i);

dU3(i)=0;  dL1(i)=-8-F2(i+1);  dL2(i)=2-F1(i+2); dL3(i)=0;

elseif i==N(k)-3    d0(i)=12-F3(i); dU1(i)=-8-F4(i);  dU2(i)=2+F1(i);

dU3(i)=0;  dL1(i)=-8-F2(i+1); dL2(i)=256-GN11; dL3(i)=840-GN1;

elseif i==N(k)-2  d0(i)=12-F3(i);  dU1(i)=-8-F4(i);  dU2(i)=2+F1(i);
```

```
dL1(i)=-936-GN12;  dL2(i)=-1440-GN2;

elseif i==N(k)-1 d0(i)=1104-GN13; dU1(i)=-424-GN14; dL1(i)=360-GN3;

elseif i==N(k)  d0(i)=240-GN4;    end, end

% The constant vector c

c1=12*r(1)+(113+G0)*a0+(60+G01)*h*a1;c2=(-2+F1(2))*a0+2*r(2);

cN1=(-168+GN102)*h^2*b2+(80+GN103)*h^3*b3+290*r(N(k)-1);

cN=4350*r(N(k))+(1080+GN02)*h^2*b2+(-3000+GN03)*h^3*b3;

C=[c1 c2 2*r(3:(N(k)-2)) cN1 cN ];

% Factorize The Matrix A=LU for 2 upper and 3 lower

U3=dU3;U2(1)=dU2(1);L0=ones(1,N(k));U1(1)=dU1(1);

U0(1)=d0(1);L1(1)=dL1(1)/U0(1);L2(1)=dL2(1)/U0(1);L3(1)=dL3(1)/U0(1)

for i=2:N(k)

if i==2   U0(i)=d0(i)-(L1(i-1)*U1(i-1)); U1(i)=dU1(i)-(L1(i-1)*U2(i-1));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)); L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i); L3(i)=dL3(i)/U0(i);

elseif  i>2 &  i<=N(k)-3 U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));U2(i)=dU2(i)-(L1(i-
1)*U3(i-1));   L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);  L3(i)=dL3(i)/U0(i);
```

```
elseif i==N(k)-2    U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));
U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));U2(i)=dU2(i)-(L1(i-1)*U3(i-
1));   L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);
elseif i==N(k)-1    U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));
U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));
L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
elseif i==N(k)    U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2);end ,end
%To solve AW=dd using factorized matrix
%Step(1):We solve LY1=dd;
y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);y1(3)=C(3)-(L2(1)*y1(1)+L1(2)*y1(2));
for i=4:N(k)
 y1(i)=C(i)-(L3(i-3)*y1(i-3)+L2(i-2)*y1(i-2)+L1(i-1)*y1(i-1));end
%Step(2):We solve UW=Y1
w(N(k))=y1(N(k))/U0(N(k));
w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-1)*w(N(k)))/U0(N(k)-1);
w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);
```

```
w(N(k)-3)=(y1(N(k)-3)-(U1(N(k)-3)*w(N(k)-2)+U2(N(k)-3)*w(N(k)-
1)+U3(N(k)-3)*w(N(k))))/U0(N(k)-3);

w(N(k)-4)=(y1(N(k)-4)-(U1(N(k)-4)*w(N(k)-3)+U2(N(k)-4)*w(N(k)-
2)+U3(N(k)-4)*w(N(k)-1)))/U0(N(k)-4);

for i=5:N(k)-1

cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2)+U3(N(k)-
i)*w(N(k)-i+3);

w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end

W1(1:N1,k)=w(2^(k-1):2^(k-1):N(k));end

% The Exact solution for y and X

for i=1:9

 h=(b-a)/10;   X(i)=a+i*h;   Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;

Ext22=(4*W1(1:N1,3)-W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;Ext31=(16*Ext22-Ext21)/15;
Ext32=(16*Ext23-Ext22)/15;Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext31'; Ext41']

plot(X,W1(1:N1,2),'red',X,Ext41,'blue',X,Y,'green')
```

```
fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f\n',MR );
```

Appendix (IV)

This appendix contains the computer programs for chapter four

Program (4.1).m:

```
% LINEAR FINITE-DIFFERENCE METHOD (4.1)
% To approximate the solution for the linear boundary-value problem
%    d5Y = F(X,Y,Y',Y''',Y''''), a<=X<=b,
% With boundary conditions Y(a) = ALPHA0, Y(b) = BETA0
,Y'(a)=ALPHA1  Y''(a)=ALPHA2 ,Y'(b)= BETA1
% INPUT:   Endpoints A,B; boundary conditions a0,a1,a2,b0,b1.
% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1
format
% Step(1):determines steps-size h and controls it
for k= 1:4
N(k)=(10*2^(k-1))-1;a=0;b=1;N1=N(1);h=(b-a)/(N(k)+1)
% The boundary conditions value
a0=1;a1=1;a2=0;b0=0;b1=-exp(2);
%The associated functions with derivative in problem and the remainder
for i=1:N(k)
```

```
x(i)=a+i*h;    p4(i)=h*(3*x(i)-1);    p3(i)=-h^2*(5*x(i)-2);
p2(i)=h^3*(3*x(i)^3-2*x(i)^2+1);    p1(i)=h^4*(2*x(i)^2);

p0(i)=h^5*(-9*x(i)^2);

r(i)=h^5*(exp(2*x(i))*(12*x(i)^4-13*x(i)^3+15*x(i)^2-56));end

%The finite-difference formula for 1st and 2nd derivative in problem

for i=2:N(k)-1

F1(i)=2*p4(i)-p3(i);  F2(i)=-8*p4(i)+2*p3(i)+2*p2(i)-p1(i);

F3(i)=12*p4(i)-4*p2(i)+2*p0(i); F4(i)=-8*p4(i)-2*p3(i)+2*p2(i)+p1(i);

F5(i)=2*p4(i)+p3(i);end

G0=76*p4(1)+351*p3(1)+36*p2(1)-18*p1(1);

G1=-432*p3(1)-72*p2(1)+36*p0(1);

G2=-108*p4(1)+81*p3(1)+36*p2(1)+18*p1(1);

G3=32*p4(1);G01=120*p4(1)+270*p3(1);G02=72*p4(1)+54*p3(1);

GN0=-565*p4(N(k))-160*p3(N(k))+60*p2(N(k))+30*p1(N(k));

GN1=-15*p4(N(k));GN2=160*p4(N(k))-20*p3(N(k));

GN3=-540*p4(N(k))+60*p2(N(k))-30*p1(N(k));

GN4=960*p4(N(k))+180*p3(N(k))-120*p2(N(k))+60*p0(N(k));

GN01=300*p4(N(k))+120*p3(N(k));

%The diagonals for matrix A for system Aw=c
```

```
d0(1)=2160-G1;dU1(1)=(-540-G2);dU2(1)=(80-G3);dL1(1)=-4320-
36*F2(2);dL2(1)=400-10*F1(3);dL3(1)=-8;dL4(1)=1;

for i=2:N(k)

if i==2  d0(i)=2160-36*F3(i);dU1(i)=-800-36*F4(i);dU2(i)=135-36*F5(i);
dL1(i)=-500-10*F2(i+1); dL2(i)=25-F1(i+2);dL3(i)=-8; dL4(i)=1;

elseif i==3  d0(i)=400-10*F3(i);dU1(i)=-175-10*F4(i);dU2(i)=32-10*F5(i)
dL1(i)=-40-F2(i+1);    dL2(i)=25-F1(i+2);    dL3(i)=-8;dL4(i)=1;

elseif i>=4 & i<N(k)-4 d0(i)=35-F3(i); dU1(i)=-16-F4(i);    U2(i)=3-F5(i);
dL1(i)=-40-F2(i+1);    dL2(i)=25-F1(i+2);    dL3(i)=-8;dL4(i)=1;

elseif i==N(k)-4    d0(i)=35-F3(i);    dU1(i)=-16-F4(i);    dU2(i)=3-F5(i);

dL1(i)=-40-F2(i+1);    dL2(i)=25-F1(i+2);    dL3(i)=-8;dL4(i)=48;

elseif i==N(k)-3    d0(i)=35-F3(i);    dU1(i)=-16-F4(i);    dU2(i)=3-F5(i);
dL1(i)=-40-F2(i+1);    dL2(i)=25-F1(i+2);    dL3(i)=-375-GN1;

elseif i==N(k)-2    d0(i)=35-F3(i);    dU1(i)=-16-F4(i);    dU2(i)=3-F5(i);

dL1(i)=-40-F2(i+1);    dL2(i)=1200-GN2;

elseif i==N(k)-1  d0(i)=35-F3(i);    dU1(i)=-16-F4(i);    dL1(i)=-2100-GN3;

elseif i==N(k)    d0(i)=2400-GN4;    end,end

% The constant vector c

c1=60*r(1)+(1700+G0)*a0+(1320+G01)*h*a1+(360+G02)*h^2*a2;

c2=(-2825+36*F1(2))*a0+72*r(2)-1860*h*a1-360*h^2*a2;
```

```
c3=157*a0+20*r(2)+60*h*a1; c4=-a0+2*r(3);

cN1=(-3+F5(N(k)-1))*b0+2*r(N(k)-1);

cN=60*r(N(k))+(1173+GN0)*b0+(-540+GN01)*h*b1;

C=[c1 c2 c3 c4 2*r(5:(N(k)-2)) cN1 cN ];

% Factorize The Matrix A=LU for 3 upper and 4 lower

U2=dU2;U1(1)=dU1(1); L0=ones(1,N(k));U1(1)=dU1(1); U0(1)=d0(1);
L1(1)=dL1(1)/U0(1); L2(1)=dL2(1)/U0(1); L3(1)=dL3(1)/U0(1);
L4(1)=dL4(1)/U0(1);

for i=2:N(k)

if i==2   U0(i)=d0(i)-(L1(i-1)*U1(i-1)); U1(i)=dU1(i)-(L1(i-1)*U2(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1))/U0(i); L4(i)=dL4(i)/U0(i);

elseif  i>2 &  i<=N(k)-4 U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)); L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-
2)*L3(i-2))/U0(i);  L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1))/U0(i); L4(i)=dL4(i)/U0(i);

elseif i==N(k)-3   U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1));  L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-
2)*L3(i-2))/U0(i); L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2))/U0(i);
```

```
L3(i)=(dL3(i)-U1(i-1)*L4(i-1))/U0(i);

elseif i==N(k)-2 U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)); L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-
2)*L3(i-2))/U0(i);  L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2))/U0(i);

elseif i==N(k)-1 U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

 U1(i)=dU1(i)-(L1(i-1)*U2(i-1)); L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-
2)*L3(i-2))/U0(i);

elseif i==N(k) U0(i)=d0(i)-U1(i-1)*L1(i-1)-L2(i-2)*U2(i-2); end,end

%To solve AW=dd using factorized matrix

%Step(1):We solve LY1=dd;

y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);

y1(3)=C(3)-(L2(1)*y1(1)+L1(2)*y1(2));

y1(4)=C(4)-(L3(1)*y1(1)+L2(2)*y1(2)+L1(3)*y1(3));

for i=5:N(k)

y1(i)=C(i)-(L4(i-4)*y1(i-4)+L3(i-3)*y1(i-3)+L2(i-2)*y1(i-2)+L1(i-1)*y1(i-
1));end

%Step(2):We solve UW=Y1

w(N(k))=y1(N(k))/U0(N(k));

w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-1)*w(N(k)))/U0(N(k)-1);
```

```
w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);

w(N(k)-3)=(y1(N(k)-3)-(U1(N(k)-3)*w(N(k)-2)+U2(N(k)-3)*w(N(k)-
1)))/U0(N(k)-3);

w(N(k)-4)=(y1(N(k)-4)-(U1(N(k)-4)*w(N(k)-3)+U2(N(k)-4)*w(N(k)-
2)))/U0(N(k)-4);

for i=5:N(k)-1

cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2);

w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end

W1(1:N1,k)=w(2^(k-1):2^(k-1):N(k));end

% The Exact solution for y and X

for i=1:9

h=(b-a)/10;    X(i)=a+i*h; Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;Ext22=(4*W1(1:N1,3)-
W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;Ext31=(16*Ext22-Ext21)/15;

Ext32=(16*Ext23-Ext22)/15;Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext31'; Ext41'];
```

```
plot(X,W1(1:N1,1),'red',X,Y,'green',X,Ext41,'blue')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f
%12.8f\n',MR );

for i=1:k

ER1(i)=max(abs(Y'-W1(:,i)));end

ERT1=[max(abs(Y'-Ext21)) max(abs(Y'-Ext31)) max(abs(Y'-Ext41)) ]
```

Program (4.2).m:

```
% LINEAR FINITE-DIFFERENCE METHOD (4.2)

% To approximate the solution for the linear boundary-value problem

%    d5Y = F(X,Y,Y',Y''',Y''''), a<=X<=b,

% With boundary conditions Y(a) = ALPHA0, Y(b) = BETA0
,Y'''(a)=ALPHA3

% Y''(b)=BETA2 ,Y''''(b)= BETA4

% INPUT:   Endpoints A,B; boundary conditions a0,a1,a2,b0,b1.

% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1

format

% Step(1):determines steps-size h and controls it

for k= 1:4  N(k)=(10*2^(k-1))-1;a=0;b=1;N1=N(1);h=(b-a)/(N(k)+1)

% The boundary conditions value
```

```
a0=1;a3=-4;b0=0;b2=-4*exp(2);b4=-32*exp(2);

%The associated functions with derivative in problem and the remainder

for i=1:N(k)

x(i)=a+i*h;  p4(i)=h*(3*x(i)-1);    p3(i)=-h^2*(5*x(i)-2);
p2(i)=h^3*(3*x(i)^3-2*x(i)^2+1);p1(i)=h^4*(2*x(i)^2);

p0(i)=h^5*(-9*x(i)^2);

r(i)=h^5*(exp(2*x(i))*(12*x(i)^4-13*x(i)^3+15*x(i)^2-56));end

%The finite-difference formula for 1st and 2nd derivative in problem

for i=2:N(k)-1

F1(i)=2*p4(i)-p3(i); F2(i)=-8*p4(i)+2*p3(i)+2*p2(i)-p1(i);

F3(i)=12*p4(i)-4*p2(i)+2*p0(i); F4(i)=-8*p4(i)-2*p3(i)+2*p2(i)+p1(i);

F5(i)=2*p4(i)+p3(i);end

G0=-90*p4(1)-140*p3(1)+210*p2(1)-105*p1(1);

G1=240*p4(1)+420*p3(1)-420*p2(1)+210*p0(1);

G2=-180*p4(1)-420*p3(1)+210*p2(1)+105*p1(1);

G3=140*p3(1);G4=30*p4(1);G03=-120*p4(1)+70*p3(1);

GN0=-14496*p4(N(k))-3624*p3(N(k))+7248*p2(N(k))+3624*p1(N(k));

GN1=7248*p4(N(k))-1812*p3(N(k));

GN2=-28992*p4(N(k))+7248*p2(N(k))-3624*p1(N(k));
```

```
GN3=36240*p4(N(k))+5436*p3(N(k))-14496*p2(N(k))+7248*p0(N(k));

GN02=7248*p4(N(k))+5436*p3(N(k));

GN04=604*p4(N(k))-1359*p3(N(k));

%The diagonals for matrix A for system Aw=c

d0(1)=-1932-G1;dU1(1)=(2856-G2);dU2(1)=(-2016-G3);dU3(1)=672-
G4;dU4(1)=-84;dL1(1)=16-F2(2);dL2(1)=-3-F1(3);dL3(1)=0;

for i=2:N(k)

if i>=2 & i<N(k)-4 d0(i)=-35-F3(i); dU1(i)=40-F4(i); dU2(i)=-25-F5(i);

dU3(i)=8;    dU4(i)=-1; dL1(i)=16-F2(i+1);dL2(i)=-3-F1(i+2);dL3(i)=0;

elseif i==N(k)-4    d0(i)=-35-F3(i);    dU1(i)=40-F4(i);    dU2(i)=-25-F5(i);
dU3(i)=8; dU4(i)=-1; dL1(i)=16-F2(i+1);dL2(i)=-46-17*F1(i+2);dL3(i)=0;

elseif i==N(k)-3 d0(i)=-35-F3(i); dU1(i)=40-F4(i);dU2(i)=-25-F5(i);
dU3(i)=8;  dL1(i)=236-17*F2(i+1);dL2(i)=-481-151*F1(i+2);

dL3(i)=-2904;

elseif i==N(k)-2    d0(i)=-484-17*F3(i);    dU1(i)=496-17*F4(i);

dU2(i)=-254-17*F5(i);    dL1(i)=2584-151*F2(i+1);    dL2(i)=10176-GN1;

elseif i==N(k)-1  d0(i)=-5526-151*F3(i);    dU1(i)=5224-151*F4(i);

dL1(i)=-11664-GN2;

elseif i==N(k)    d0(i)=4416-GN3;    end,end
```

```
% The constant vector c

c1=210*r(1)+(-504+G0)*a0+(-168+G03)*h^3*a3;
c2=(3+F1(2))*a0+2*r(2);cN3=b0+2*r(N(k)-3);

cN2=-52*b0+6*h^4*b4+34*r(N(k)-2);

cN1=(1801+151*F5(N(k)-1))*b0-660*h^2*b2+124*h^4*b4+302*r(N(k)-1);

cN=7248*r(N(k))+(24+GN0)*b0+(1440+GN02)*h^2*b2+(-
4224+GN04)*h^4*b4;

C=[c1 c2 2*r(3:(N(k)-4)) cN3 cN2 cN1 cN ];

% Factorize The Matrix A=LU for 4 upper and 3 lower

U4=dU4;L1=ones(1,N(k));

U1(1)=dU1(1);U0(1)=d0(1);L1(1)=dL1(1)/U0(1);L2(1)=dL2(1)/U0(1);

L3(1)=dL3(1)/U0(1);U2(1)=dU2(1);U3(1)=dU3(1);

for i=2:N(k)

if i==2  U0(i)=d0(i)-(L1(i-1)*U1(i-1));  U1(i)=dU1(i)-(L1(i-1)*U2(i-1));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1));  U3(i)=dU3(i)-(L1(i-1)*U4(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i); L3(i)=dL3(i)/U0(i);

elseif i>2 &i<=N(k)-3 U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));
```

```
U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);  L3(i)=dL3(i)/U0(i);

elseif i==N(k)-2  U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);

elseif i==N(k)-1

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);

elseif i==N(k)

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3));end,end

%To solve AW=dd using factorization of  matrix

%Step(1):We solve LY1=dd;

y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);
```

```
y1(3)=C(3)-(L2(1)*y1(1)+L1(2)*y1(2));

for i=4:N(k)

y1(i)=C(i)-(L3(i-3)*y1(i-3)+L2(i-2)*y1(i-2)+L1(i-1)*y1(i-1));end

%Step(2):We solve UW=Y1

w(N(k))=y1(N(k))/U0(N(k));

w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-1)*w(N(k)))/U0(N(k)-1);

w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);

w(N(k)-3)=(y1(N(k)-3)-(U1(N(k)-3)*w(N(k)-2)+U2(N(k)-3)*w(N(k)-
1)+U3(N(k)-3)*w(N(k))))/U0(N(k)-3);

w(N(k)-4)=(y1(N(k)-4)-(U1(N(k)-4)*w(N(k)-3)+U2(N(k)-4)*w(N(k)-
2)+U3(N(k)-4)*w(N(k)-1)+U4(N(k)-4)*w(N(k))))/U0(N(k)-4);

for i=5:N(k)-1

cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2)+U3(N(k)-
i)*w(N(k)-i+3)+U4(N(k)-i)*w(N(k)-i+4);

w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end

W1(1:N1,k)=w(2^(k-1):2^(k-1):N(k));end

% The Exact solution for y and X

for i=1:9

h=(b-a)/10;    X(i)=a+i*h;    Y(i)=(1-X(i))*exp(2*X(i));end
```

```
% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;

Ext22=(4*W1(1:N1,3)-W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;Ext31=(16*Ext22-Ext21)/15;

Ext32=(16*Ext23-Ext22)/15;Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext31'; Ext41'];

plot(X,W1(1:N1,1),'red',X,Y,'green',X,Ext41,'blue')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f
%12.8f\n',MR );

for i=1:k

ER1(i)=max(abs(Y'-W1(:,i)));end

ERT1=[max(abs(Y'-Ext21)) max(abs(Y'-Ext31)) max(abs(Y'-Ext41)) ]
```

Program (4.3).m:

```
% LINEAR FINITE-DIFFERENCE METHOD (4.3)

% To approximate the solution for the linear boundary-value problem

%    d6Y = F(X,Y,Y',Y''',d4Y,d5Y), a<=X<=b,

% With boundary conditions Y(a) = ALPHA0, Y(b) = BETA0
,Y'(a)=ALPHA1
```

```
% Y'(b)=BETA1,Y"(a)= ALPHA2 ,Y"(b)=BETA2

% INPUT:   Endpoints A,B; boundary conditions a0,a1,a2,b0,b1,b2.

% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1

format

% Step(1): determines steps-size h and controls it

for k= 2:5N(k)=(10*2^(k-1))-1;a=0;b=1;N1=9;h=(b-a)/(N(k)+1)

% The boundary conditions value

a0=1;a1=1;a2=0;b0=0;b1=-exp(2);b2=-4*exp(2);

%The associated functions with derivative in problem and the remainder

for i=1:N(k)    x(i)=a+i*h;    p5(i)=2*h*x(i)^2;    p4(i)=h^2*(-5*x(i)^3);

p3(i)=-h^3*(3*x(i)^2-1); p2(i)=h^4*(10*x(i)+3);

p1(i)=-h^5*(3+5*x(i)-x(i)^2); p0(i)=-h^6*(2*x(i)-x(i)^2);

r(i)=h^6*(exp(2*x(i))*(-80*x(i)^4-37*x(i)^3+110*x(i)^2-43*x(i)-121));

end

%The finite-difference formula for 1st and 2nd derivative in problem

for i=3:N(k)-2   F1(i)=-p5(i);   F2(i)=4*p5(i)+2*p4(i)-p3(i);

F3(i)=-5*p5(i)-8*p4(i)+2*p3(i)+2*p2(i)-p1(i);

F4(i)=12*p4(i)-4*p2(i)+2*p0(i);
```

```
F5(i)=5*p5(i)-8*p4(i)-2*p3(i)+2*p2(i)+p1(i);

F6(i)=-4*p5(i)+2*p4(i)+p3(i);F7(i)=p5(i);end

G0=-11900*p5(1)+532*p4(1)+2457*p3(1)+252*p2(1)-126*p1(1);

G1=15120*p5(1)-3024*p3(1)+504*p2(1)+252*p0(1);

G2=-3780*p5(1)-756*p4(1)+567*p3(1)+252*p2(1)+126*p1(1);

G3=560*p5(1)+224*p4(1);G01=-9240*p5(1)+840*p4(1)+1890*p3(1);

G02=-2520*p5(1)+504*p4(1)+378*p3(1);

G20=-14125*p5(2)+180*(2*p4(2)-p3(2));

G21=-21600*p5(2)+180*(-8*p4(2)+2*p3(2)+2*p2(2)-p1(2));

G22=-10800*p5(2)+180*(12*p4(2)-4*p2(2)+2*p0(2));

G23=-4000*p5(2)+180*(-8*p4(2)-2*p3(2)+2*p2(2)+p1(2));

G24=675*p5(2)+180*(2*p4(2)+p3(2));

G201=9300*p5(2);G202=1800*p5(2);

GN20=-14125*p5(N(k)-1)+180*(2*p4(N(k)-1)+p3(N(k)-1));

GN24=21600*p5(N(k)-1)+180*(-8*p4(N(k)-1)-2*p3(N(k)-1)+2*p2(N(k)-
1)+p1(N(k)-1));

GN23=-10800*p5(N(k)-1)+180*(12*p4(N(k)-1)-4*p2(N(k)-1)+2*p0(N(k)-
1));
```

```
GN22=4000*p5(N(k)-1)+180*(-8*p4(N(k)-1)+2*p3(N(k)-1)+2*p2(N(k)-
1)-p1(N(k)-1));

GN21=-675*p5(N(k)-1)+180*(2*p4(N(k)-1)-p3(N(k)-1));

GN201=9300*p5(N(k)-1);GN202=-1800*p5(N(k)-1);

GN0=11900*p5(N(k))+532*p4(N(k))-
2457*p3(N(k))+252*p2(N(k))+126*p1(N(k));

GN3=-15120*p5(N(k))+3024*p3(N(k))+504*p2(N(k))+252*p0(N(k));

GN2=3780*p5(N(k))-756*p4(N(k))-567*p3(N(k))+252*p2(N(k))-
126*p1(N(k));

GN1=-560*p5(N(k))+224*p4(N(k));GN01=-9240*p5(N(k))-
840*p4(N(k))+1890*p3(N(k));

GN02=2520*p5(N(k))+504*p4(N(k))-378*p3(N(k));

%The diagonals for matrix A for system Aw=c

d0=[11016-G1 -13500-G22 -40*ones(1,N(k)-4)-F4(3:N(k)-2) -13500-
GN23 11016-GN3];

dU1=[-17280-G2 9600-G23 30*ones(1,N(k)-4)-F5(3:N(k)-2) 12960-
GN24];

dU2=[15280-G3 -4050-G24 -12*ones(1,N(k)-4)-F6(3:N(k)-2)];

dU3=[-8100 864 2*ones(1,N(k)-5)-F7(3:N(k)-3)];

dU4=[2376 -60 zeros(1,N(k)-6)];dU5=[-296 zeros(1,N(k)-6)];
```

```
dL1=[12960-G21 30*ones(1,N(k)-4)-F3(3:N(k)-2) 9600-GN22 -17280-
GN2];

dL2=[-12*ones(1,N(k)-4)-F2(3:N(k)-2) -4050-GN21 15280-GN1];

dL3=[2*ones(1,N(k)-5)-F1(4:N(k)-2) 864 -8100 ];dL4=seqreverse(dU4);

dL5=seqreverse(dU5);

% The constant vector c

c1=252*r(1)+(2996+G0)*a0+G01*h*a1+(-720+G02)*h^2*a2;

c2=(5814+G20)*a0+(2520+G201)*h*a1+G202*h^2*a2+360*r(2);

c3=(-2+F1(3))*a0+2*r(3);cN2=(-2+F7(N(k)-2))*b0+2*r(N(k)-2);

cN1=(5814+GN20)*b0+(-2520+GN201)*h*b1+GN202*h^2*b2+360*r(N(k)-1);

cN=252*r(N(k))+(2996+GN0)*b0+(-720+GN02)*h^2*b2+GN01*h*b1;

C=[c1 c2 c3 2*r(4:(N(k)-3)) cN2 cN1 cN ];

% Factorize The Matrix A=LU for 4 upper and 3 lower

U5=dU5;L1=ones(1,N(k));U1(1)=dU1(1);U0(1)=d0(1);L1(1)=dL1(1)/U0(1
);L2(1)=dL2(1)/U0(1);L3(1)=dL3(1)/U0(1);U2(1)=dU2(1);
U3(1)=dU3(1);U4(1)=dU4(1);L4(1)=dL4(1)/U0(1);L5(1)=dL5(1)/U0(1);

for i=2:N(k)

if i==2 U0(i)=d0(i)-(L1(i-1)*U1(i-1));U1(i)=dU1(i)-(L1(i-1)*U2(i-1));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)); U3(i)=dU3(i)-(L1(i-1)*U4(i-1));
```

```
U4(i)=dU4(i)-(L1(i-1)*U5(i-1)); L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);
L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);
L3(i)=(dL3(i)-U1(i-1)*L4(i-1))/U0(i);
L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);  L5(i)=dL5(i)/U0(i);
elseif i==3   U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));
U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));
U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2));
U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));
U4(i)=dU4(i)-(L1(i-1)*U5(i-1));
L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);
L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2))/U0(i);
L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);
L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);   L5(i)=dL5(i)/U0(i);
elseif i==4
U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3));
U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3));
U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));
U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));
```

```
U4(i)=dU4(i)-(L1(i-1)*U5(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);
L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);L5(i)=dL5(i)/U0(i);

elseif i>=5 &i<=N(k)-5

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));
U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1));L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-
2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i); L5(i)=dL5(i)/U0(i);

 elseif i==N(k)-4

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));
```

```
U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));
U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);

elseif i==N(k)-3

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-4)*U4(i-4));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);
```

```
L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

elseif i==N(k)-2

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

elseif i==N(k)-1

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

elseif i==N(k)

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));   end,end

%To solve AW=dd using factorized matrix
```

```
%Step(1):We solve LY1=dd;

y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);y1(3)=C(3)-
(L2(1)*y1(1)+L1(2)*y1(2));

y1(4)=C(4)-(L3(1)*y1(1)+L2(2)*y1(2)+L1(3)*y1(3));

y1(5)=C(5)-(L4(1)*y1(1)+L3(2)*y1(2)+L2(3)*y1(3)+L1(4)*y1(4));

for i=6:N(k)

y1(i)=C(i)-(L5(i-5)*y1(i-5)+L4(i-4)*y1(i-4)+L3(i-3)*y1(i-3)+L2(i-2)*y1(i-
2)+L1(i-1)*y1(i-1)); end

%Step(2):We solve UW=Y1

w(N(k))=y1(N(k))/U0(N(k));w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-
1)*w(N(k)))/U0(N(k)-1);

w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);

w(N(k)-3)=(y1(N(k)-3)-(U1(N(k)-3)*w(N(k)-2)+U2(N(k)-3)*w(N(k)-
1)+U3(N(k)-3)*w(N(k))))/U0(N(k)-3);

w(N(k)-4)=(y1(N(k)-4)-(U1(N(k)-4)*w(N(k)-3)+U2(N(k)-4)*w(N(k)-
2)+U3(N(k)-4)*w(N(k)-1)+U4(N(k)-4)*w(N(k))))/U0(N(k)-4);

w(N(k)-5)=(y1(N(k)-5)-(U1(N(k)-5)*w(N(k)-4)+U2(N(k)-5)*w(N(k)-
3)+U3(N(k)-5)*w(N(k)-2)+U4(N(k)-5)*w(N(k)-1))+U5(N(k)-
5)*w(N(k)))/U0(N(k)-5);

for i=6:N(k)-1
```

```
cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2)+U3(N(k)-
i)*w(N(k)-i+3)+U4(N(k)-i)*w(N(k)-i+4)+U5(N(k)-i)*w(N(k)-i+5);

w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end

W1(1:N1,k-1)=w(2^(k-1):2^(k-1):N(k));end

% The Exact solution for y and X

for i=1:9   h=(b-a)/10;  X(i)=a+i*h; Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;Ext22=(4*W1(1:N1,3)-
W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;Ext31=(16*Ext22-Ext21)/15;

Ext32=(16*Ext23-Ext22)/15;Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext31'; Ext41'];

plot(X,W1(1:N1,2),'red',X,Y,'green',X,Ext21,'blue')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f
%12.8f\n',MR );

for i=1:k-1   ER1(i)=max(abs(Y'-W1(:,i)));end,ER1

ERT1=[max(abs(Y'-Ext21)) max(abs(Y'-Ext31)) max(abs(Y'-Ext41)) ]
```

Program (4.4).m:

```
% LINEAR FINITE-DIFFERENCE METHOD (4.4)

% To approximate the solution for the linear boundary-value problem

%    d6Y = F(X,Y,Y',Y''',d4Y,d5Y), a<=X<=b,

% With boundary conditions Y(a) = ALPHA0, Y(b) = BETA0
,Y'''(a)=ALPHA3

% Y''(b)=BETA2 ,d5Y(a)= ALPHA5 ,d4Y(b)=BETA4

% INPUT:   Endpoints A,B; boundary conditions a0,a3,a5,b0,b2,b4.

% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1

format

% Step(1):determines steps-size h and controls it

for k= 2:5

N(k)=(10*2^(k-1))-1;a=0;b=1;N1=9;h=(b-a)/(N(k)+1)

% The boundary conditions value

a0=1;a3=-4;a5=-48;b0=0;b2=-4*exp(2);b4=-32*exp(2);

%The associated functions with derivative in problem and the remainder

for i=1:N(k)   x(i)=a+i*h;  p5(i)=2*h*x(i)^2;  p4(i)=h^2*(-5*x(i)^3);

p3(i)=-h^3*(3*x(i)^2-1);   p2(i)=h^4*(10*x(i)+3);

p1(i)=-h^5*(3+5*x(i)-x(i)^2);  p0(i)=-h^6*(2*x(i)-x(i)^2);
```

```
r(i)=h^6*(exp(2*x(i))*(-80*x(i)^4-37*x(i)^3+110*x(i)^2-43*x(i)-121));
end
%The finite-difference formula for 1st and 2nd derivative in problem
for i=3:N(k)-2  F1(i)=-p5(i); F2(i)=4*p5(i)+2*p4(i)-p3(i);
F3(i)=-5*p5(i)-8*p4(i)+2*p3(i)+2*p2(i)-p1(i);
F4(i)=12*p4(i)-4*p2(i)+2*p0(i);
F5(i)=5*p5(i)-8*p4(i)-2*p3(i)+2*p2(i)+p1(i);
F6(i)=-4*p5(i)+2*p4(i)+p3(i); F7(i)=p5(i);end
G0=150*p5(1)-100*p4(1)-100*p3(1)+150*p2(1)-75*p1(1);
G1=-540*p5(1)+300*p4(1)+300*p3(1)-300*p2(1)+150*p0(1);
G2=720*p5(1)-300*p4(1)-300*p3(1)+150*p2(1)+75*p1(1);
G3=-420*p5(1)+100*p4(1)+100*p3(1);G4=90*p5(1);
G03=60*p5(1)-100*p4(1)+50*p3(1);G05=45*p5(1)+25*p4(1)-50*p3(1);
G20=100*p5(2)+25*(2*p4(2)-p3(2));
G21=-360*p5(2)+25*(-8*p4(2)+2*p3(2)+2*p2(2)-p1(2));
G22=480*p5(2)+25*(12*p4(2)-4*p2(2)+2*p0(2));
G23=-280*p5(2)+25*(-8*p4(2)-2*p3(2)+2*p2(2)+p1(2));
G24=60*p5(2)+25*(2*p4(2)+p3(2));G203=40*p5(2);G205=-20*p5(2);
```

GN20=-113463*p5(N(k)-1)+9513*(2*p4(N(k)-1)+p3(N(k)-1));

GN24=329112*p5(N(k)-1)+9513*(-8*p4(N(k)-1)-2*p3(N(k)-1)+2*p2(N(k)-1)+p1(N(k)-1));

GN23=-348138*p5(N(k)-1)+9513*(12*p4(N(k)-1)-4*p2(N(k)-1)+2*p0(N(k)-1));

GN22=162792*p5(N(k)-1)+9513*(-8*p4(N(k)-1)+2*p3(N(k)-1)+2*p2(N(k)-1)-p1(N(k)-1));

GN21=-30303*p5(N(k)-1)+9513*(2*p4(N(k)-1)-p3(N(k)-1));

GN202=41580*p5(N(k)-1);GN204=-7812*p5(N(k)-1);

GN0=-168*p5(N(k))-101472*p4(N(k))-25368*p3(N(k))+50736*p2(N(k))+25368*p1(N(k));

GN4=30912*p5(N(k))+253680*p4(N(k))+38052*p3(N(k))-101472*p2(N(k))+50736*p0(N(k));

GN3=-81648*p5(N(k))-202944*p4(N(k))+50736*p2(N(k))-25368*p1(N(k));

GN2=71232*p5(N(k))+50736*p4(N(k))-12684*p3(N(k));

GN1=-20328*p5(N(k));

GN02=-10080*p5(N(k))+50736*p4(N(k))+38052*p3(N(k));

GN04=29568*p5(N(k))+4228*p4(N(k))-9513*p3(N(k));

%The diagonals for matrix A for system Aw=c

```
d0=[540-G1 180-G22 -40*ones(1,N(k)-4)-F4(3:N(k)-2) -509625-GN23
594336-GN4];

dU1=[-1170-G2 -280-G23 30*ones(1,N(k)-4)-F5(3:N(k)-2) 358776-
GN24];

dU2=[1320-G3 240-G24 -12*ones(1,N(k)-4)-F6(3:N(k)-2)];

dU3=[-810-G4 -108 2*ones(1,N(k)-5)-F7(3:N(k)-3)];

dU4=[252 20 zeros(1,N(k)-6)];dU5=[-30 zeros(1,N(k)-6)];

dL1=[-60-G21 30*ones(1,N(k)-4)-F3(3:N(k)-2) 386560-GN22 -1224912-
GN3];

dL2=[-12*ones(1,N(k)-4)-F2(3:N(k)-2) -163080-GN21 1323968-GN2];

dL3=[2*ones(1,N(k)-5)-F1(4:N(k)-2) 32616 -790032-GN1];

dL4=[zeros(1,N(k)-6) -1661 246432];dL5=[zeros(1,N(k)-6) -31408];

% The constant vector c

c1=150*r(1)+(102+G0)*a0+G03*h^3*a3+(72+G05)*h^5*a5;

c2=(-8+G20)*a0+G203*h^3*a3+(12+G205)*h^5*a5+50*r(2);

c3=(-2+F1(3))*a0+2*r(3);cN2=(-2+F7(N(k)-2))*b0+2*r(N(k)-2);

cN1=(103586+GN20)*b0+(-
27180+GN202)*h^2*b2+GN204*h^4*b4+19026*r(N(k)-1);

cN=50736*r(N(k))+(118384+GN0)*b0+(GN02)*h^2*b2+(-
28992+GN04)*h^4*b4;
```

```
C=[c1 c2 c3 2*r(4:(N(k)-3)) cN2 cN1 cN ];

% Factorize The Matrix A=LU for 3 upper and 3 lower

U5=dU5;L1=ones(1,N(k));

U1(1)=dU1(1);U0(1)=d0(1);L1(1)=dL1(1)/U0(1);L2(1)=dL2(1)/U0(1);

L3(1)=dL3(1)/U0(1);U2(1)=dU2(1);U3(1)=dU3(1);U4(1)=dU4(1);

L4(1)=dL4(1)/U0(1);L5(1)=dL5(1)/U0(1);

for i=2:N(k)

  if i==2  U0(i)=d0(i)-(L1(i-1)*U1(i-1));  U1(i)=dU1(i)-(L1(i-1)*U2(i-1));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1));   U3(i)=dU3(i)-(L1(i-1)*U4(i-1));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1)); L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);  L5(i)=dL5(i)/U0(i);

elseif i==3    U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1));
```

```
 L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i); L5(i)=dL5(i)/U0(i);

elseif i==4

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i); L5(i)=dL5(i)/U0(i);

elseif i>=5 &i<=N(k)-5

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));
```

```
U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

 L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i); L5(i)=dL5(i)/U0(i);

 elseif i==N(k)-4

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1));

 L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);
```

```
L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);

elseif i==N(k)-3

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

elseif i==N(k)-2

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);
```

```
L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

elseif i==N(k)-1

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

elseif i==N(k)

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));   end,end

%To solve AW=dd using factorized matrix

%Step(1):We solve LY1=dd;

y1(1)=C(1);    y1(2)=C(2)-L1(1)*y1(1);

y1(3)=C(3)-(L2(1)*y1(1)+L1(2)*y1(2));

y1(4)=C(4)-(L3(1)*y1(1)+L2(2)*y1(2)+L1(3)*y1(3));

y1(5)=C(5)-(L4(1)*y1(1)+L3(2)*y1(2)+L2(3)*y1(3)+L1(4)*y1(4));

for i=6:N(k)

 y1(i)=C(i)-(L5(i-5)*y1(i-5)+L4(i-4)*y1(i-4)+L3(i-3)*y1(i-3)+L2(i-
2)*y1(i-2)+L1(i-1)*y1(i-1));end
```

```
%Step(2):We solve UW=Y1

w(N(k))=y1(N(k))/U0(N(k));w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-
1)*w(N(k)))/U0(N(k)-1);

w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);

w(N(k)-3)=(y1(N(k)-3)-(U1(N(k)-3)*w(N(k)-2)+U2(N(k)-3)*w(N(k)-
1)+U3(N(k)-3)*w(N(k))))/U0(N(k)-3);

w(N(k)-4)=(y1(N(k)-4)-(U1(N(k)-4)*w(N(k)-3)+U2(N(k)-4)*w(N(k)-
2)+U3(N(k)-4)*w(N(k)-1)+U4(N(k)-4)*w(N(k))))/U0(N(k)-4);

w(N(k)-5)=(y1(N(k)-5)-(U1(N(k)-5)*w(N(k)-4)+U2(N(k)-5)*w(N(k)-
3)+U3(N(k)-5)*w(N(k)-2)+U4(N(k)-5)*w(N(k)-1))+U5(N(k)-
5)*w(N(k)))/U0(N(k)-5);

for i=6:N(k)-1

cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2)+U3(N(k)-
i)*w(N(k)-i+3)+U4(N(k)-i)*w(N(k)-i+4)+U5(N(k)-i)*w(N(k)-i+5);

w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end

W1(1:N1,k-1)=w(2^(k-1):2^(k-1):N(k));end

% The Exact solution for y and X

for i=1:9    h=(b-a)/10;    X(i)=a+i*h;    Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;
```

```
Ext22=(4*W1(1:N1,3)-W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;Ext31=(16*Ext22-Ext21)/15;

Ext32=(16*Ext23-Ext22)/15;Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext31'; Ext41'];

plot(X,W1(1:N1,1),'red',X,W1(1:N1,2),'yellow',X,Y,'green',X,Ext41,'blue')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f
%12.8f\n',MR );

for i=1:k-1     ER1(i)=max(abs(Y'-W1(:,i)));   end

ERT1=[max(abs(Y'-Ext21)) max(abs(Y'-Ext31)) max(abs(Y'-Ext41)) ]
```

Program (4.5).m:

```
% LINEAR FINITE-DIFFERENCE METHOD (4.5)

% To approximate the solution for the linear boundary-value problem

   d7Y = F(X,Y,Y',Y'",d4Y,d5Y,d6Y), a<=X<=b,

% With boundary conditions Y(a) = ALPHA0, Y(b) = BETA0
,Y'(a)=ALPHA1

% Y"(a)=ALPHA2,d5Y(a)=ALPHA5 ,Y'(a)= BETA1 ,d5Y(b)=BETA5

% INPUT:   Endpoints A,B; boundary conditions a0,a1,a2,a5,b0,b1,b5.

% OUTPUT:  Approximations W(I) TO Y(X(I)) for each i=0,1,...,N+1
```

```
format
% Step(1)
% determines step-size h and controls it
for k= 2:5
N(k)=(10*2^(k-1))-1;a=0;b=1;N1=9;
h=(b-a)/(N(k)+1)
% The boundary conditions value
a0=1;a1=1;a2=0;a5=-48;b0=0;b1=-exp(2);b5=-80*exp(2);
%The associated functions with derivative in problem and the remainder
for i=1:N(k)
x(i)=a+i*h;
p6(i)=h*x(i)^2;  p5(i)=-h^2*(2*x(i)^2-1);
p4(i)=-h^3*(x(i)^3+x(i)^2+1);    p3(i)=h^4*(5*x(i)^3);
p2(i)=h^5*(9*x(i));    p1(i)=-h^6*(1-x(i)); p0(i)=-h^7*(3+2*x(i));
r(i)=h^7*(4*exp(2*x(i)))*(6*x(i)^4-3*x(i)^3+13*x(i)^2-29*x(i)-71);end
%The finite-difference formula for 1st and 2nd derivative in problem
for i=3:N(k)-2
   F1(i)=2*p6(i)-p5(i);
```

F2(i)=-12*p6(i)+4*p5(i)+2*p4(i)-p3(i);

F3(i)=30*p6(i)-5*p5(i)-8*p4(i)+2*p3(i)+2*p2(i)-p1(i);

F4(i)=-40*p6(i)+12*p4(i)-4*p2(i)+2*p0(i);

F5(i)=30*p6(i)+5*p5(i)-8*p4(i)-2*p3(i)+2*p2(i)+p1(i);

F6(i)=-12*p6(i)-4*p5(i)+2*p4(i)+p3(i);

F7(i)=2*p6(i)+p5(i);end

G10=133875*p6(1)-472500*p5(1)+63560*p4(1)+73710*p3(1)+7560*p2(1)-3780*p1(1);

G11=-207360*p6(1)+609120*p5(1)-60480*p4(1)-90720*p3(1)-15120*p2(1)+7560*p0(1);

G12=106920*p6(1)-165240*p5(1)-7560*p4(1)+17010*p3(1)+7560*p2(1)+3780*p1(1);

G13=-40320*p6(1)+30240*p5(1)+4480*p4(1);

G14=6885*p6(1)-1620*p5(1);

G101=86940*p6(1)-362880*p5(1)+62160*p4(1)+56700*p3(1);

G102=16200*p6(1)-97200*p5(1)+25200*p4(1)+11340*p3(1);

G105=-3888*p6(1)-864*p5(1)+1008*p4(1);

G20=-138960*p6(2)+124272*p5(2)+6120*(2*p4(2)-p3(2));

G21=306720*p6(2)-233280*p5(2)+6120*(-8*p4(2)+2*p3(2)+2*p2(2)-p1(2));

G22=-311040*p6(2)+164160*p5(2)+6120*(12*p4(2)-4*p2(2)+2*p0(2));

G23=210240*p6(2)-63360*p5(2)+6120*(-8*p4(2)-2*p3(2)+2*p2(2)+p1(2));

G24=-79920*p6(2)+6480*p5(2)+6120*(2*p4(2)+p3(2));

G25=12960*p6(2)+1728*p5(2);G201=-60480*p6(2)+60480*p5(2);

G205=-864*p6(2)-1584*p5(2);

GN11=1260*p6(N(k)-1)-168*p5(N(k)-1);

GN12=-7770*p6(N(k)-1)-630*p5(N(k)-1)+595*(2*p4(N(k)-1)-p3(N(k)-1));

GN13=20440*p6(N(k)-1)+6160*p5(N(k)-1)+595*(-8*p4(N(k)-1)+2*p3(N(k)-1)+2*p2(N(k)-1)-p1(N(k)-1));

GN14=-30240*p6(N(k)-1)-15960*p5(N(k)-1)+595*(12*p4(N(k)-1)-4*p2(N(k)-1)+2*p0(N(k)-1));

GN15=29820*p6(N(k)-1)+22680*p5(N(k)-1)+595*(-8*p4(N(k)-1)-2*p3(N(k)-1)+2*p2(N(k)-1)+p1(N(k)-1));

GN10=-13510*p6(N(k)-1)-12082*p5(N(k)-1)+595*(2*p4(N(k)-1)+p3(N(k)-1));

GN101=5880*p6(N(k)-1)+5880*p5(N(k)-1);

GN105=84*p6(N(k)-1)-154*p5(N(k)-1);

GN1=1512*p6(N(k)); GN2=5670*p6(N(k))-16065*p5(N(k));

GN3=-55440*p6(N(k))+85680*p5(N(k))+28560*p4(N(k))-7140*p3(N(k));

GN4=143640*p6(N(k))-192780*p5(N(k))-128520*p4(N(k))+21420*p2(N(k))-10710*p1(N(k));

```
GN5=-
204120*p6(N(k))+257040*p5(N(k))+257040*p4(N(k))+64260*p3(N(k))-
42840*p2(N(k))+21420*p0(N(k));

GN0=108738*p6(N(k))-133875*p5(N(k))-157080*p4(N(k))-
57120*p3(N(k))+21420*p2(N(k))+10710*p1(N(k));

GN01=-52920*p6(N(k))+64260*p5(N(k))+85680*p4(N(k))+42840*p3(N(k));

GN05=12096*p6(N(k))+8568*p5(N(k))+4284*p4(N(k))-3213*p3(N(k));

%The diagonals for matrix A for system Aw=c

d0=[552960-G11 -963900-G21 -4880-35*F4(3) -14700-105*F4(4) -
126*ones(1,N(k)-6)-F4(5:N(k)-2) -124950-GN14 -1566720-GN5];

dU1= [-564840-G12 380800-G23  2640-35*F5(3) 7938-105*F5(4)
70*ones(1,N(k)-6)-F5(5:N(k)-2) 91630-GN15];

dU2=[402240-G13 -80325-G24 -816-35*F6(3) -2450-105*F6(4) -
22*ones(1,N(k)-6)-F6(5:N(k)-2)];

dU3=[-177120-G14  -G25  110-35*F7(3) 330-105*F7(4) 3*ones(1,N(k)-
7)-F7(5:N(k)-3) ];

dU4=[43200  2380 zeros(1,N(k)-6)]; dU5=[-4440 zeros(1,N(k)-6)];

dL1=[2056320-G21 5730-35*F3(3) 17150-105*F3(4) 140*ones(1,N(k)-6)-
F3(5:N(k)-2) 124950-GN13 1600380-GN4];

dL2=[-4800-35*F2(3)  -13230-105*F2(4) -98*ones(1,N(k)-6)-F2(5:N(k)-
2) -83300-GN12  -1139680-GN3];
```

```
dL3=[7350-105*F1(4)  42*ones(1,N(k)-6)-F1(5:N(k)-2) 34986-GN11
501840-GN2];

dL4=[-10*ones(1,N(k)-6) -8330 -122400-GN1];

dL5=[ones(1,N(k)-7)  850   12580];

% The constant vector c

c1=(252000+G10)*a0+(110880+G101)*h*a1+G102*h^2*a2-(5616-
G105)*h^5*a5+7560*r(1);

c2=(1395275+G20)*a0+(963900+G201)*h*a1+(214200)*h^2*a2+G205*h
^5*a5+12240*r(2);

c3=(-2016+35*F1(3))*a0-(840)*h*a1+12*h^5*a5+70*r(3);

c4=2388*a0+840*h*a1+210*r(4);

c5=-a0+2*r(5);

cN2=(-3+F7(N(k)-2))*b0+2*r(N(k)-2);

cN1=(35836+GN10)*b0-(14280-
GN101)*h*b1+GN105*h^5*b5+1190*r(N(k)-1);

cN=(-714000+GN0)*b0+(314160+GN01)*h*b1 -(15912-
GN05)*h^5*b5+21420*r(N(k));

C=[c1 c2 c3 c4 c5 2*r(6:(N(k)-3)) cN2 cN1 cN ];

% Factorize The Matrix A=LU for 5 upper and 5 lower

U5=dU5;L1=ones(1,N(k));
```

```
U1(1)=dU1(1);U0(1)=d0(1);L1(1)=dL1(1)/U0(1);L2(1)=dL2(1)/U0(1);
L3(1)=dL3(1)/U0(1);U2(1)=dU2(1);U3(1)=dU3(1);U4(1)=dU4(1);
L4(1)=dL4(1)/U0(1);L5(1)=dL5(1)/U0(1);
for i=2:N(k)
   if i==2
    U0(i)=d0(i)-(L1(i-1)*U1(i-1)); U1(i)=dU1(i)-(L1(i-1)*U2(i-1));
    U2(i)=dU2(i)-(L1(i-1)*U3(i-1));U3(i)=dU3(i)-(L1(i-1)*U4(i-1));
    U4(i)=dU4(i)-(L1(i-1)*U5(i-1));
    L1(i)=(dL1(i)-U1(i-1)*L2(i-1))/U0(i);
    L2(i)=(dL2(i)-U1(i-1)*L3(i-1))/U0(i);
    L3(i)=(dL3(i)-U1(i-1)*L4(i-1))/U0(i);
      L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);  L5(i)=(dL5(i))/U0(i);
      elseif i==3
      U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2));
      U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2));
      U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2));
      U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));
      U4(i)=dU4(i)-(L1(i-1)*U5(i-1));
```

```
L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i); L5(i)=(dL5(i))/U0(i);

elseif i==4

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);

L5(i)=(dL5(i))/U0(i);

elseif i==5

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4));
```

```
U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i); L5(i)=(dL5(i))/U0(i)

        elseif i>5 &i<=N(k)-6

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4)+L5(i-5)*U5(i-5))

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

 U4(i)=dU4(i)-(L1(i-1)*U5(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);
```

```
L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);  L5(i)=(dL5(i))/U0(i);

    elseif i==N(k)-5

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4)+L5(i-5)*U5(i-5));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);  L5(i)=(dL5(i))/U0(i);

        elseif i==N(k)-4

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4)+L5(i-5)*U5(i-5));
```

```
U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

U4(i)=dU4(i)-(L1(i-1)*U5(i-1));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

L4(i)=(dL4(i)-U1(i-1)*L5(i-1))/U0(i);

        elseif i==N(k)-3

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4)+L5(i-5)*U5(i-5));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

U3(i)=dU3(i)-(L1(i-1)*U4(i-1)+L2(i-2)*U5(i-2));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);
```

```
L3(i)=(dL3(i)-U1(i-1)*L4(i-1)-U2(i-2)*L5(i-2))/U0(i);

    elseif i==N(k)-2

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4)+L5(i-5)*U5(i-5));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

U2(i)=dU2(i)-(L1(i-1)*U3(i-1)+L2(i-2)*U4(i-2)+L3(i-3)*U5(i-3));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

L2(i)=(dL2(i)-U1(i-1)*L3(i-1)-U2(i-2)*L4(i-2)-U3(i-3)*L5(i-3))/U0(i);

elseif i==N(k)-1

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4)+L5(i-5)*U5(i-5));

U1(i)=dU1(i)-(L1(i-1)*U2(i-1)+L2(i-2)*U3(i-2)+L3(i-3)*U4(i-3)+L4(i-
4)*U5(i-4));

L1(i)=(dL1(i)-U1(i-1)*L2(i-1)-U2(i-2)*L3(i-2)-U3(i-3)*L4(i-3)-U4(i-
4)*L5(i-4))/U0(i);

    elseif i==N(k)

U0(i)=d0(i)-(L1(i-1)*U1(i-1)+L2(i-2)*U2(i-2)+L3(i-3)*U3(i-3)+L4(i-
4)*U4(i-4)+L5(i-5)*U5(i-5));    end,end

 %To solve AW=dd using factorized matrix
```

```
%Step(1):We solve LY1=dd;
y1(1)=C(1);y1(2)=C(2)-L1(1)*y1(1);y1(3)=C(3)-(L2(1)*y1(1)+L1(2)*y1(2));
y1(4)=C(4)-(L3(1)*y1(1)+L2(2)*y1(2)+L1(3)*y1(3));
y1(5)=C(5)-(L4(1)*y1(1)+L3(2)*y1(2)+L2(3)*y1(3)+L1(4)*y1(4));
y1(6)=C(6)-
(L5(1)*y1(1)+L4(2)*y1(2)+L3(3)*y1(3)+L2(4)*y1(4)+L1(5)*y1(5));
for i=7:N(k)
y1(i)=C(i)-(L5(i-5)*y1(i-5)+L4(i-4)*y1(i-4)+L3(i-3)*y1(i-3)+L2(i-2)*y1(i-
2)+L1(i-1)*y1(i-1));end
%Step(2):We solve UW=Y1
w(N(k))=y1(N(k))/U0(N(k));w(N(k)-1)=(y1(N(k)-1)-U1(N(k)-
1)*w(N(k)))/U0(N(k)-1);
w(N(k)-2)=(y1(N(k)-2)-(U1(N(k)-2)*w(N(k)-1)+U2(N(k)-
2)*w(N(k))))/U0(N(k)-2);
w(N(k)-3)=(y1(N(k)-3)-(U1(N(k)-3)*w(N(k)-2)+U2(N(k)-3)*w(N(k)-
1)+U3(N(k)-3)*w(N(k))))/U0(N(k)-3);
w(N(k)-4)=(y1(N(k)-4)-(U1(N(k)-4)*w(N(k)-3)+U2(N(k)-4)*w(N(k)-
2)+U3(N(k)-4)*w(N(k)-1)+U4(N(k)-4)*w(N(k))))/U0(N(k)-4);
w(N(k)-5)=(y1(N(k)-5)-(U1(N(k)-5)*w(N(k)-4)+U2(N(k)-5)*w(N(k)-
3)+U3(N(k)-5)*w(N(k)-2)+U4(N(k)-5)*w(N(k)-1))+U5(N(k)-
5)*w(N(k)))/U0(N(k)-5);
```

```
for i=6:N(k)-1

cc(N(k)-i)=U1(N(k)-i)*w(N(k)-i+1)+U2(N(k)-i)*w(N(k)-i+2)+U3(N(k)-
i)*w(N(k)-i+3)+U4(N(k)-i)*w(N(k)-i+4)+U5(N(k)-i)*w(N(k)-i+5);

w(N(k)-i)=(y1(N(k)-i)-cc(N(k)-i))/U0(N(k)-i);end

W1(1:N1,k-1)=w(2^(k-1):2^(k-1):N(k));end

% The Exact solution for y and X

for i=1:9

   h=(b-a)/10;    X(i)=a+i*h;    Y(i)=(1-X(i))*exp(2*X(i));end

% The Extrapolation for approximated solution

Ext21=(4*W1(1:N1,2)-W1(1:N1,1))/3;

Ext22=(4*W1(1:N1,3)-W1(1:N1,2))/3;

Ext23=(4*W1(1:N1,4)-W1(1:N1,3))/3;

Ext31=(16*Ext22-Ext21)/15;Ext32=(16*Ext23-Ext22)/15;

Ext41=(64*Ext32-Ext31)/63;

% The approximated solution

MR=[X; Y;W1(:,1)';W1(:,2)';W1(:,3)';W1(:,4)';Ext31'; Ext41'];

plot(X,W1(1:N1,1),'red',X,Y,'green',X,W1(1:N1,2),'blue')

fprintf(1,'%6.1f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f %12.8f\n',MR );

for i=1:k-1
```

```
ER1(i)=max(abs(Y'-W1(:,i)));end ,ER1

ERT1=[max(abs(Y'-Ext21)) max(abs(Y'-Ext31)) max(abs(Y'-Ext41)) ]
```

References:

[1] Aftabizaded ,A.R, Wiener, Joseph, 1986, **Existence And Uniqueness Theorems For Third Order Boundary Value Problems**, Rendiconti del Seminario Metematico della a Universita di Padova.

[2] Burden, R.L , Faires, J.D,2001, **Numerical Analysis**, 7th ed, Brooks/Cole.

[3] Dulabh, Yogash,1992,**Finite Difference Methods For Higher-Order Boundary-Value and Eigen value Problems**, Department of Mathematics and Statistics , Brunel University, Uxbridge.

[4] http://en.wikipedia.org/wiki/Finite_difference_method

[5] LeVeque, Rundall.J, 2006 , **Finite Difference Methods For Differential Equations**, University of Washington

[6] MA, Ruyun, March 1998, **Existence And Uniqueness Theorems For Some Fourth-Order Nonlinear Boundary Value Problems**, Internet, J.Math. & Math. Sci, Hindawi Publishing Crop.

[7] Mathews, J.H, 1992, **Matlab Programming Guidebook for Numerical Methods**, 2nd .ed , Enlewood cliffs, New Jersey 07632.

[8] Urroz, Gilberto.E, 2001, **Ordinary Differential Equations With SCILAB**, Info Clearinghouse .Com.

[9] Erturk,Vedat,Momani,Shaher,2008,**DIFFERNTIAL TRANSFORM TECHNIUE FOR SOLVING FIFTH ORDER BOUNDARY VALUE PROBLEMS**, Mathematical and computational Application, Vol.13,No.2,pp.113-121,©Association for Scientific Research.

جامعة النجاح الوطنية

كلية الدراسات العليا

طرق عددية لحل

المعادلات التفاضلية الحدودية ذات الدرجات العليا

اعداد

بسمة عثمان عبد المحسن العزة

اشراف

د.سمير مطر

قدمت هذه الأطروحة استكمالا لمتطلبات درجة الماجستير في الرياضيات المحوسبة بكلية الدراسات العليا بجامعة النجاح الوطنية في نابلس، فلسطين .

2009

الملخص

المعادلات التفاضلية الحدودية واسعة الاستخدام في مجالات متعدد في الحياة و في الدراسات العلمية، ولقد عملت دراسات كثيرة بمناقشتها وعرض طرقا متعددة لحلها و ذاك بشكل خاص لدرجة الثانية منها مع وجود بعض الدراسات و الطرق التي ناقشت أنواع خاصة بدرجات أعلى من هذه المشكلات.

أما في أطروحتنا هذه فقد قمنا بـ:

1- دراسة و حل معادلات حدية خطية ذات درجات عليا من الدرجة الثالثة و حتى السابعة و ذلك بشكلها العام و استخدمنا طريقة الفروق الدقيقة للتوصل إلى نظام من المعادلات خطية بسيطة.
2- حل النظام بطريقة(LU-decomposition) لتقليل العمليات الحسابية.
3- رفعنا درجة الدقة للطريقة و ذلك من خلال استخدام طريقة (Richardson's Extrapolation Method) للحصول على نتائج أكثر دقة دون الحاجة لتقليل طول الفترات الجزئية في المسألة.
4- دراسة معادلة تفاضلية حدية غير خطية خاصة من الدرجة الثامنة و طرق حلها.
5- تطوير بعض الطرق العددية لتكون قادرة على حل أي معادلة تفاضلية غير خطية من نفس النوع لأي درجة زوجية أقل من ثمانية و ذلك من خلال تصميم برنامج حاسبي بلغة (MATLAB 7.0) للتعامل و الحصول على معاملات نظام حل هذه المعادلات.

و لقد أوضحت الدراسة أن:

- هذه الطريقة المتبعة في البحث تعطي نتائج جيدة.
- الخطأ يزداد عندما تكون درجة المعادلة عالية لاسيما أنها تعتمد على جميع المشتقات التي تسبقها – فمثلا المعادلة التفاضلية الخامسة تعتمد على المشتقة الأولى حتى الرابعة – مما يجعل خطأ التقريب متراكما.
- هذه الطريقة تعتمد على القيم الحدية المعطاة فعند غياب القيمة الحدية الابتدائية لطرفي الفترة يصبح الخطأ أعلى.

- نوع القيمة الحدية المعطاة في المسألة يؤثر على دقة الحل و الخطأ. فيما إما إذا كانت القيمة الحدية عند مشتقة زوجية، فردية، قريبة أو بعيدة.
- حل درجات عليا من المعادلات التفاضلية بهذه الطريقة يحتاج إلى معادلات طويلة تزداد بازدياد الدرجة وإلى جهد وحسابات مطولة يصعب عملها لمرات متكررة إلا باستخدام طرق خاصة على الحاسب.