Journal of Basrah Researches ((Sciences)) Volume 37.Number 4 .D ((2011))

Available online at: www.basra-science-journal.org

ISSN –1817 –2695

On Solution of Two Point Second Order Boundary Value Problems using Semi-Analytic Method

Luma. N. M.Tawfiq **&** Khalid. M. M. Al-Abrahemee
Department of Mathematics, College of Education-Ibn Al-Haitham, University of Baghdad .

Abstract

In this paper a new method is proposed for the solution of two-point second order boundary-value problems(TPBVP) ,that is, we concerned with constructing polynomial solutions to two point second order boundary value problems for ordinary differential equation.

A semi-analytic technique using two-point osculatory interpolation with the fit equal numbers of derivatives at the end points of an interval [0,1] is compared with conventional methods via a series of examples and is shown to be that seems to converge faster and more accurately than the conventional methods and generally superior, particularly for problems involving nonlinear equations and/or boundary conditions.

Also we introduce some general observations about control of a residual and control of the true error and we prove, there is a more useful connection between scaled residual and true error.

1. Introduction

The most general form of the problem to be considered is :

$$y'' = f(x, y, y'), \quad x \in [a, b],$$

with boundary conditions : $y(a) = A$, $y(b) = B$

there is no loss in generality in taking a = 0 and b = 1, and we will sometimes employ this slight simplification. We view f as a generally nonlinear function of y and y', but for the present, we will take f = f(x) only. For such a problem to have a solution it is generally necessary either that $f(x) \neq 0$ hold, or that $A \neq 0$ at one or both ends of the interval. When $f(x) \equiv 0$, and A = 0 , B =0 the BVP is said to be homogeneous and will in general have only the trivial solution, $y(x) \equiv 0$ [1].In this paper we introduce a new technique for the qualitative and quantitative analysis of non homogeneous linear TPBVP using two-point polynomial interpolation .

2. Approximation Theory

The primary aim of a general approximation is to represent non-arithmetic quantities by arithmetic quantities so that the accuracy can be ascertained to a desired degree. Secondly, we are also concerned with the amount of computation required to achieve this accuracy. A complicated function f(x) usually is approximated by an easier function of the form $\varphi(x; a_0, \dots, a_n)$ where $a_0, \dots, a_n$ are parameters to be determined so as to characterize the best approximation of f.

In this paper, we shall consider only the interpolatory approximation. From Weierstrass Approximation Theorem ,it follows that one can always find a polynomial that is arbitrarily close to a given function on some finite interval. This means that the approximation error is bounded and can be reduced by the choice of the adequate polynomial. Unfortunately Weierstrass Approximation Theorem is not a constructive one, i.e. it does not present a way how to obtain such a polynomial. i.e. the interpolation problem can also be formulated in another way, viz. as the answer to the following question: How to find a .good. representative of a function that is not known explicitly, but only at some points of the domain of interest .In this paper we use Oscillatory Interpolation since has high order with the same given points in the domain .

2.1. Osculatory Interpolation[2]

Given $\{x_i\}$, i = 1, . . .k and values $f_i^{(0)}, \dots, f_i^{(r_i)}$,where r_i are nonnegative integers and $f_i = f(x_i)$.We want to construct a polynomial P(x) such that $P^{(j)}(x_i) = f_i^{(j)}$ …………… (1), for i = 1, . . . , k and j = 0, . . . , r_i.
Such a polynomial is said to be an sculatory interpolating polynomial of a function f

Remark

The degree of P(x) is at most $\sum_{i=1}^{k}(r_i+1)-1$.

In this paper we use two-point oscillatory interpolation [2]. Essentially this is a generalization of interpolation using Taylor polynomials and for that reason oscillatory interpolation is sometimes referred to as two-point Taylor interpolation. The idea is to approximate a function y(x) by a polynomial P(x) in which values of y(x) and any number of its derivatives at given points are fitted by the corresponding function values and derivatives of P(x).

In this paper we are particularly concerned with fitting function values and derivatives at the two end points of a finite interval, say [0,1], wherein a useful and succinct way of writing a osculatory interpolant $P_{2n+1}(x)$ of degree 2n + 1 was given for example by Phillips [3] as :

$$P_{2n+1}(x) = \sum_{j=0}^{n} \{y^{(j)}(0)\, q_j(x) + (-1)^j\, y^{(j)}(1)\, q_j(1-x)\} \quad \ldots\ldots\ldots\ldots(2)$$

$$q_j(x) = (x^j/j!)(1-x)^{n+1} \sum_{s=0}^{n-j} \binom{n+s}{s} x^s = Q_j(x)/j! \quad \ldots\ldots\ldots\ldots(3)$$

so that (2) with (3) satisfies :

$$y^{(r)}(0) = P_{2n+1}^{(r)}(0)\ , \quad y^{(r)}(1) = P_{2n+1}^{(r)}(1)\ , \quad r=0,1,2,\ldots,n.$$

implying that $P_{2n+1}(x)$agrees with the appropriately truncated Taylor series for y(x) about x = 0 and x = 1.The error on [0, 1] is given by :

$$R_{2n+1} = y(x) - P_{2n+1}(x) = \frac{(-1)^{n+1} x^{(n+1)} (1-x)^{n+1} y^{(2n+2)}(\varepsilon)}{(2n+2)!} \quad \text{where } 0 \langle \varepsilon \langle 1 \text{ and}$$

$\mathbf{y}^{(2n+2)}$ is assumed to be continuous.

The osculatory interpolant for $P_{2n+1}(x)$ may converge to y(x) in [0,1] irrespective of whether the intervals of convergence of the constituent series intersect or are disjoint .The important consideration here is whether $R_{2n+1} \to 0$ as $n\to\infty$ for all x in [0,1]. In the application to the boundary value problems in this paper such convergence with n is always confirmed numerically .We observe that (2) fits an equal number of derivatives at each end point but it is possible and indeed sometimes desirable to use polynomials which fit different numbers of derivatives at the end points of an interval. As an example of a two-point osculatory interpolant we may take n = 2 so that (2) with (3) becomes the quintic :

$$P_5(x)= (1-x)^3(1+3x+6x^2)y(0) + x^3(10-15x+6x^2)y(1) + x(1-x)^3(1+3x)y'(0) - x^3(1-x)(4-3x)\, y'(1) + 1/2x^2(1-x)^3 y''(0) + 1/2x^3(1-x)^2 y''(1)$$

Satisfying :

$P_5(0)=y(0)$, $P'_5(0)= y'(0)$, $P''_5(0)= y''(0)$.

$P_5(1)=y(1)$, $P'_5(1)= y'(1)$, $P''_5(1)= y''(1)$.

Finally we observe that (2) can be written directly in terms of the Taylor coefficients a_i and b_i about x = 0 and x = 1 respectively, as :

$$P_{2n+1}(x) = \sum_{j=0}^{n} \{ a_j Q_j(x) + (-1)^j b_j Q_j(1-x) \} \qquad \text{....}\ (4)$$

3. Solution Of Two Point Second-Order Boundary Value Problems

We consider the boundary value problem

$$y'' + f(x ,y ,y') = 0 \qquad \text{..................}\ (5)$$

$$g_i(y(0) , y(1) , y'(0), y'(1)) = 0 , \quad i = 1 , 2 \quad \text{.......}\ (6)$$

where f , g_1, g_2 are nonlinear functions of their arguments and g_1 and g_2 are given in three kinds [4] :

1- $y(0) = a_0$, $y(1) = b_0$ …… (6a), and we say this kind Dirichlet condition (value specified).

2- $y'(0) = a_1$,$y'(1) = b_1$ … .(6b) , and we say this kind Neumann condition (Derivative specified).

3- $c_0y'(0)+c_1y(0) = a$, $d_0y'(1) + d_1y(1) = b$ …. (6c), where c_0 ,c_1 ,d_0 ,d_1 are all positive constants not all are zero but c_1 , d_0 are equal to zero or c_0 , $d_1$are equal to zero and we say this kind Mixed condition (Gradient & value) .

The simple idea behind the use of two-point polynomials is to replace y(x) in problem (5)–(6), or an alternative formulation of it, by a P_{2n+1} which enables any unknown boundary values or derivatives of y(x) to be computed . The first step therefore is to construct the P_{2n+1} . To do this we need the Taylor coefficients of y (x) at x = 0 :

$$y = a_0 + a_1 x + \sum_{i=2}^{\infty} a_i x^i \qquad \text{........}\ (7a)$$

into (5)and equate coefficients of powers of x yields the system of equations can be solved to obtain a_i (a_0 , a_1)for all $i \geq 2$. Also we need the Taylor coefficients of y (x) at x = 1. Using MATLAB throughout we simply insert the series forms :

$$y = b_0 + b_1(x-1) + \sum_{i=2}^{\infty} b_i(x-1)^i \quad \text{........}(7b)$$

into (5) and equate coefficients of powers of (x−1). In order to obtain b_i (b_0, b_1) for all $i \geq 2$.The notation implies that the coefficients depend only on the indicated unknowns a_0, a_1, b_0, b_1. The algebraic manipulations is needed for this process .We are now in a position to construct a $P_{2n+1}(x)$ from (7) of the form (2) and use it as a replacement in the problem (5)–(6). Since we have only the four unknowns to compute for any n we only need to generate two equations from this procedure as two equations are already supplied by the boundary conditions (6). An obvious way to do this would be to satisfy the equation (5) itself at two selected points $x = c_1$, $x = c_2$ in [0,1] so that the two required equations become :

$$P''_{2n+1}(c_i) + f \{ P_{2n+1}(c_i) , P'_{2n+1}(c_i), c_i \}= 0 , \quad i=1,2. \quad \text{.............}(8)$$

An alternative approach is to recast the problem in an integral form before doing the replacement. Extensive computations have shown that this generally provides a more accurate polynomial representation for a given n. We therefore use this alternative formulation throughout this thesis although we should keep in mind that the procedure based on (8) is a viable option and shares many common features with the approach outlined below. Of the many ways we could provide an integral formulation we adopt the following. We first integrate (5) to obtain :

$$y'(x) - a_1 + \int_0^x f(y(s), y'(s), s)\, ds = 0 \qquad \text{………………….(9)}$$

and again to find :

$$y(x) - a_0 - x\, a_1 + \int_0^x (x-s)\, f(y(s), y'(s), s)\, ds = 0 \qquad \text{……………….(10)}$$

where $a_0 = y(0)$ and $a_1 = y'(0)$. Putting $x = 1$ in (9) and (10) then gives :

$$b_1 - a_1 + \int_0^1 f(y(s), y'(s), s)\, ds = 0 \qquad \text{……………………….(11)}$$

and

$$b_0 - a_0 - a_1 + \int_0^1 (1-s)\, f(y(s), y'(s), s)\, ds = 0 \qquad \text{………………….(12)}$$

where $b_0 = y(1)$ and $b_1 = y'(1)$.

The precise way we make the replacement of $y(x)$ with a $P_{2n+1}(x)$ in (11) and (12) depends on the nature of $f(y, y', x)$ and will be explained in the examples which follow. In any event the important point to note is that once this replacement has been made, the equations (6), (11) and (12) constitute the four equations we require to determine the set $\{a_0, b_0, a_1, b_1\}$. As we shall see the fact that the number of unknowns is independent of the number of derivatives fitted represents perhaps the most important feature of the method.

We make the following points at this stage :

(i) In the majority of cases where the boundary conditions are simple enough the system of algebraic equations may be reduced a priori to a system in two unknowns, since the boundary condition can be substituted directly into the integral formulations (11) and (12), which MATLAB can be utilized to solve. That is, if we have the BC(6a), then we have only the unknown pair $\{a_1, b_1\}$ and is known the required polynomial can be constructed. For the benefit of the reader the entire procedure for Examples in section 4 .And if, we have the BC(6b), then we have only the unknown pair $\{a_0, b_0\}$and is known the required polynomial can be constructed. Also if, we have the BC(6c), then we have only the unknown pair $\{a_0, b_1\}$or $\{a_1, b_0\}$ and is known the required polynomial can be constructed .

(ii) The method offers a certain amount of flexibility. For example we could choose to satisfy (9) and (10) at two internal points or we could use alternative integral formulations. The fact remains that whatever strategy we adopt produces a quickly convergent sequence of values of the set $\{a_0, a_1, b_0, b_1\}$ as n increases.

(iii) Throughout we assess the accuracy of the procedure by examining the convergence with n. Using a symbolic computational facility such as MATLAB, computing the required convergent is not an issue. Where possible we can also run checks on our solutions using shooting with MATLAB codes.

(iv) We compare our method with the other method. We now consider a number of examples designed to illustrate the convergence, accuracy, implementation and utility of the method. In what follows the use of bold digits in the tables is intended to give a rough visual indication of the convergence.

Remark

1- All computations in the following examples were performed in the MATLAB environment, Version 7, running on a Microsoft Windows 2003 Professional operating system .

2- In the following examples when analytical solutions are known so that we can measure the error of a solution. When analytical solutions are not known, we compare our results to values computed by other methods .

4. Examples

In this section we introduce some examples illustrates suggested method :

Linear boundary value problems (BVPs) can be used to model several physical phenomena. For example, a common problem in civil engineering concerns the deflection of a beam of rectangular cross section subject to uniform loading, while the ends of the beam are supported so that they undergo no deflection. This problem is linear second-order TPBVP[5] .Now, we give many other examples, we first consider the linear problem with Dirichlet BC :

Example 1

$$y'' - 4(y - x) = 0 \quad , \quad y(0) = 0 \quad , \quad y(1) = 2 \qquad \ldots \ (13)$$

has exact solution [6] : $e^2 (e^4 - 1)^{-1} (e^{2x} - e^{-2x}) + x$

Here (11) and (12) become :

$$b_1 - a_1 + 2 - 4\int_0^1 y(s)\,ds = 0 \qquad \ldots\ldots\ldots.. \quad (14)$$

$$8/3 - a_1 - 4\int_0^1 (1-s)\,y(s)\,ds = 0 \qquad \ldots\ldots\ldots. \quad (15)$$

and the coefficients : a_2 ,b_2, a_3 , b_3 ,…can be found from (7a) and (7b) .

Abinitio inclusion of the boundary conditions in (13) has reduced the number of unknowns to two, namely $\{a_1, b_1\}$, which are computed by solving (14) and (15) with y(s) replaced by a $P_{2n+1}(s)$. The results for n = 2, 3, 4 are displayed in Table 1. We can see that there is clear convergence with n to the ‘exact’ values which are obtained using MATLAB boundary value software. Table 2 gives the compare between the suggested method and other methods and figure 1 gives the accuracy of the method.

TABLE 1: The result of the methods for n = 2, 3, 4 of example1

		P5	**P7**	**P9**	
a1		1.5511387164	1.5514458006	1.5514410832	
b1		3.0749482402	3.0746246085	3.0746294890	
X	**Y:exact**	**P5**	**P7**	**P9**	**|Y-P9|**
0.25	0.3936766919	0.3937912461	0.3936753464	0.3936767011	0.000000009170200
0.5	0.8240271368	0.8244047619	0.8240204194	0.8240272117	0.000000074822647
0.75	1.3370861339	1.3372355396	1.3370844322	1.3370861455	0.000000011556981
S.S.E = 5.81608482230229E-15					

Then from table 1 and the relation (2)and (3) in the previous section we have :

$P_5 = .121739\, x^5 - .662526e^{-1}\, x^4 + .393375\, x^3 + 1.55114\, x$

$P_7 = .114177e^{-1}\, x^7 - .905686\, e^{-2}\, x^6 + .804532\, e^{-1}\, x^5 - .189035\, e^{-2}\, x^4 + .367631\, x^3 + 1.55145\, x$

$P_9 = .628069e^{-3}\, x^9 - .652397e^{-3}\, x^8 + .774834\, e^{-2}\, x^7 - .403156\, e^{-3}\, x^6 + 0.0736107\, x^5 + 0.367627\, x^3 + 1.55144\, x$

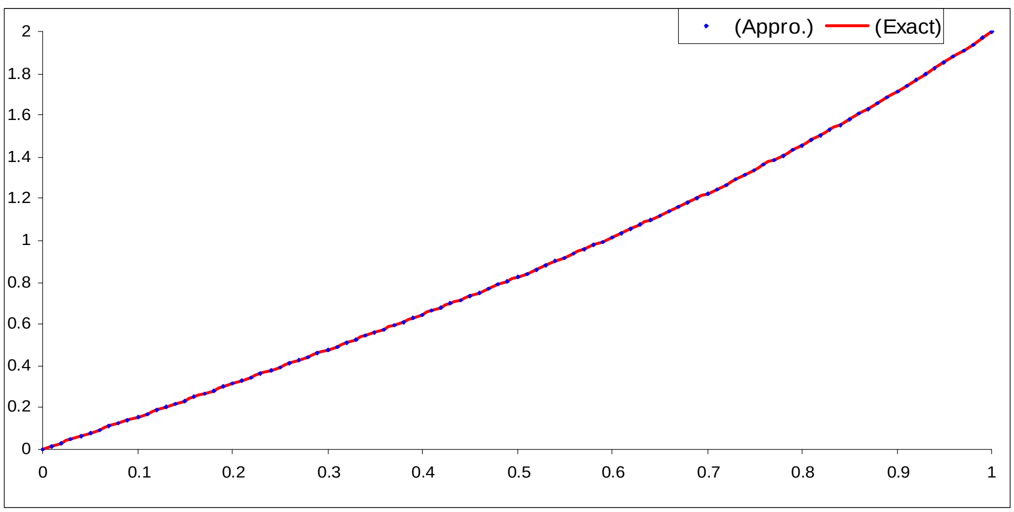

Figure1:Comparison between the exact solution and semi-analytic method P_9

Now we give the comparison between the solution of suggested method and solution of other methods in the following table :

TABLE 2: A Comparison between P_9 and other methods of example 1

x	**Y**	**Φ_1 by using linear shooting method**	**Φ_2 by sing linear Finite-Difference method**	**P9 by using Oscillatory interpolation**	**|Y-P9|**
0.25	0.393676692	0.3936767	0.39367669	0.3936767011	0.0000000092
0.5	0.824027137	0.8240271	0.82402714	0.8240272117	0.0000000748
0.75	1.337086134	1.337086	1.33708613	1.3370861455	0.0000000116

Now, we give the nonlinear problem with Neumann boundary conditions :

Example 2

$y'' = y^3 - y\ y'$ with BC : $y'(0) = -1$, $y'(1) = -1/4$

have the exact solution [6] : $y(x) = 1 / (x + 1)$

The result of method given in the following table :

Table 3 : The result of the methods for n = 2, 3, 4 of example 2

		P5	P7	P9	
a0		-2.0730192434	-1.9094622257	1.0084460394	
b0		2.2375726588	2.0790662118	0.5109116568	
x	**Y**	**P5**	**P7**	**P9**	**| Y-P9 |**
0.25	0.8000000000	0.7788980098	0.7715765789	0.7715767601	0.028423239916215
0.5	0.6666666667	0.6509123549	0.6567887010	0.6618496652	0.004817001507875
0.75	0.5714285714	0.6070021412	0.6088950935	0.6025318078	0.031103236348971
S.S.E = 0.00179849538224153					

Then from table 3 and the relation (2) and (3) in the previous section we have :

$\mathbf{P_9 = 21.1466\ x^9 - 95.0579\ x^8 + 163.2064\ x^7 - 127.9210\ x^6 + 39.2950\ x^5 - 1/6\ x^3 - x + 1.0084}$

The accuracy of the solution given in the following figure :

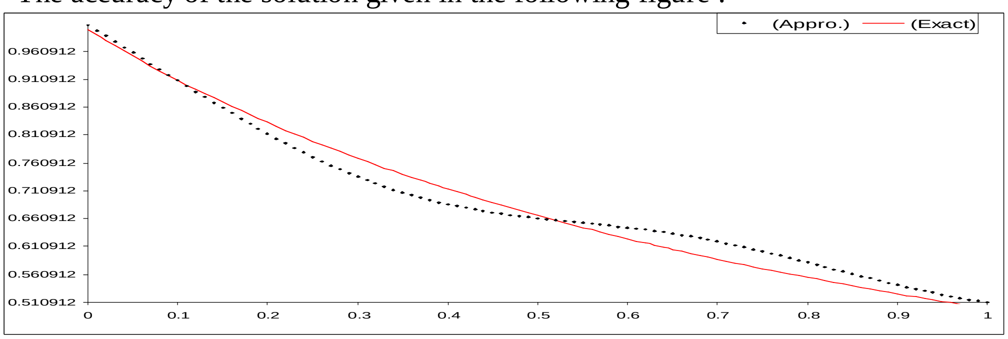

Figure2: A comparison between exact and approximate solution of example 2

5. Error Estimation And Control

In this section we begin with some general observation about control of a residual and control of the true error and we prove that if the scaled residual is less than a given tolerance , then the true error is also less than the tolerance. And if the B.V.P is well - conditioned , a small residual implies a small true error , but this need not be true if the B.V.P is ill-conditioned [7]. A practical distinction is that the method we consider approximate r(x) (residual) to a lower order than e(x) (true error) .Furthermore ,the solution y(x) is the natural weight when controlling a norm of e(x) and y'(x) is the natural weight when controlling a norm of r(x) .

We describe a new B.V.P solver that controls a residual and the true error .We consider method suggested in this thesis that approximate the solution of y(x) of (5) - (6) on a mesh: $0 = x_0 < x_1 < \dots\dots < x_{N+1} = 1$ by a function P(x) that is smooth on subinterval $[x_i , x_{i+1}]$.The mesh spacing $h_i = x_{i+1} - x_i$ and convergence is the considered as : $h = \max_i h_i$ tented to zero. We assume that the true error e(x) =

P(x) - y(x) and a natural measure of error is the residual in the differential equations [8] : $r(x) = P''(x) - f(x, P(x), P'(x))$ ……….(16)

This can be interpolated as saying that P(x) is the exact solution of the problem(5)- (6) with is data f(x, y, y') and g(y(a),y(b)) perturbed by residual , e,g,

$$P''(x) = (x ,P(x) , P'(x)) + r(x)$$

<u>5.1. Residual</u>

In this subsection we introduce some details about how controls the size of the residual . The residual is scaled so that it has same order of convergence as the true error [9]. Residual control has important virtues : residual are well-defined no matter how bad the approximate solution , and residuals can be evaluated any where simple by evaluating : f(x , P(x), P'(x)) or g(P(0) , P(1)).

Now the approximate solution P(x) is smooth on subintervals $[x_i , x_{i+1}]$, so the size of the residual on the subinterval is measured by using a weighted norm $|r(x)|$ at each x and defining $\| r(x) \|_i = \underset{x_i \le x \le x_{i+1}}{Max} |r(x)|$.For a give tolerance ϵ , the aims to produce solution for which $\max_i \| r(x) \| \le \epsilon$.

In this constructs a mesh that approximately equidistributes the residual. It might seem that controlling the scaled residual , $h_i \| r(x) \|_i$, is obviously less demanding than controlling the residual , $\| r(x) \|_i$, because of the small factor h_i , but this neglects the role of the norm. Now , subtracting (5) from (16) lead to :

$$r(x) = P''(x) - y''(x) - [f(x ,P(x) , P'(x)) – f(x, y(x),y'(x))] \qquad \text{………(17)}$$

With the usual assumption that f satisfies a Lipschitz condition :

$$| f(x , P(x), P'(x)) - f(x ,y(x), y'(x)) | \le L |P(x) - y(x) |$$

We assume that $e(x) = P(x) – y(x)$ is $O(h^{n+1})$ and $P'(x) – y'(x)$ is $O(h^{n})$, so $P''(x) – y''(x)$ is $O(h^{n-1})$, hence the last term in (17) is $O(h^{n+1})$, so to leading order the residual is equal to the error in the 2_{nd} derivative .This implies that the scaled residual is $O(h^{n+1})$, the same as the true error. Now , if the residuals are uniformly small, P(x) is a good solution in the sense that it is the exact solution of a problem close to the one supplied to the solver. Further , for reasonable well-condition problem , small residuals imply that P(x) is close to y(x) , even when h is not small enough [10] that the (n+1) order convergence is evident .

We prove now ,the there is a more useful connection between scaled residua and true error. To investigate the relationship between scaled residual and true error ,we begin by integrating(17)over a subinterval of $[x_i , x]$,where $x_i < x \le x_{i+1}$

$$\int_{xi}^{x} r(x)\, dx = e'(x) – e'(x_i) - \int_{xi}^{x} [f(x ,P(x), P'(x)) – f(x, y(x),y'(x)]dx$$

where : $e' = P' – y'$

Again integrate over $[x_i , \beta]$, where $\beta \in (x_i , x_{i+1}]$, we get :

$$\int_{xi}^{B} (x-s)r(s)ds = e(\beta) – e(x_i) – e'(x_i)(\beta–x_i) - \int_{xi}^{B} (x-s)[f(s,P(s),P'(s))– f(s, y(s),y'(s))]d \qquad \text{……………….(18)}$$

Suppose now that the method of order n is super convergent at mesh point , meaning that if the method is of order n , a norm of the error at mesh points is at least $O(h^{n+1})$ [9] , so that $e(x_i)$ is $O(h^{n+1})$ and $e'(x_i) = P'(x_i) - y'(x_i)$ is $O(h^{n})$ and $|e(\beta)| = \| e(x) \|_i$. As argued earlier , the integrand on the right hand side of (18) is $O(h^{n+1})$ and the interval is of length no bigger than h_i , so

$$\int_{xi}^{B} \| (x-s)\, r(s)\, ds \| = \| e(x) \|_{i.} + O(h^{n+1})$$

Then we have the inequality : $\int_{xi}^{B} \| (x\text{-}s)\, r(s)\, ds \| \leq h_i \ \| r(s) \|_{i.}$

Then the size of the scaled residual is an upper bound on the size of the true error. Now , if we require that $\max_i h_i \|r(s)\|_{i.} \leq \epsilon$, for a tolerance ϵ , then we have $\max_i \|e(x)\|_{i.} \leq \epsilon$, this is a strong argument for controlling the size of the residual .

5.2 . Error Estimates

The error on [0,1] is given by :

$$e_n = P_n(x) - y(x) = \frac{(-1)^{n+1} x^{(n+1)} (1-x)^{n+1} y^{(2n+2)}(\xi)}{(2n+1)!}$$

where $0 < \zeta < 1$ and $y^{(2n+2)}$ is assumed to be continuous .The Osculator interpolant for $P_n(x)$ may converge to $y(x)$ in [0 , 1] irrespective of whether the intervals of convergence of the constituent series intersect or are disjoint. The important consideration her is whether $e_n \to 0$ as $n \to \infty$ for all x in [0 , 1].In the application to the BVP´s in this thesis such convergence with n is always confirmed numerically .

References

[1] J. M. McDonough , Lectures In Basic Computational Numerical Analysis , Departments of Mechanical Engineering and Mathematics, University of Kentucky , 2007 .

[2] V. Girault and L. R. Scott, Hermite Interpolation of Non smooth Functions Preserving Boundary Conditions ,Mathematics of Computation, Vol. 71, No.239, pp.1043-1074, 2002.

[3] G .M.Phillips , "Explicit forms for certain Hermite approximations ",BIT 13 (1973) 177-180.

[4] R. J. LeVeque, Finite Difference Methods for Differential Equations , University of Washington ,2006 .

[5] S. M.Mahmoud, Quintic Spline Collocation Methods for the Solution of Linear Second-Order Boundary Value Problems ,Tishreen University Journal for Studies and Scientific Research ,Vol.29, No.1 ,,2007.

[6] L. R. Burden and J. D. Faires , "**Numerical Analysis**" , Seventh Edition, 2001.

[7] N .HALE and D.R.MOORE,A Sixth-Order Extension to the MATLAB Packge bvp4c of J.A. kierzenka and L. Shmpine , Oxford University Computing Laboratory , Numerical Analysis Group, Oxford, England.

[8] E. S . Pav , Numerical Methods Course Notes ,University of California, 2005.

[9] J.A. kierzenka and L , F. Shampine , A BVP solver that Controls Residual and Error , Southern Methodist University , Dollas ,U,S.A, 2007.

[10] I. Gladwell , Boundary value problem , Scholarpedia,3(1) 2853, 2008, Southern Methodist University , Dallas ,TX.