Partition Method for Solving Boundary Value Problem Using B-Spline Functions

Bushra E. Kashem*

Received on:2/12/2008

Accepted on:2/4/2009

Abstract

This paper is concerned with the approximated solution of linear two-points boundary value problem (LTPBVP) using Partition method with the aid of B-Spline functions as basis functions. The result of this method is compared with the exact solution. Two numerical examples are given for conciliated the results of this method.

Keywords:- Boundary Value Problem (BVP), linear two-points boundary value problem (LTPBVP), B-Spline Function (BS),

طريقة التجزئة لحل مسألة القيمة الحدودية بأستخدام الدوال التوصيلية

الخلاصة

هذا البحث اهتم بالحل التقريبي لمسألة القيم الحدودية الخطية بين نقطتين بأستخدام طريقة التجزئة مع الدوال التوصيلية كدوال اساسية.النتائج لهذه الطريقة تم مقارنتها مع الحل الحقيقي.اعطي مثالين عددين لحساب النتائج بهذه الطريقة.

1. Introduction

A number of problems in engineering and physics give rise to a linear two-points boundary value problem, such that the formulation of model for particular physical situation, effects and deformation of shells [9].

A general form of linear two-points boundary value problem:- [7]

$$L[y] = y^{(n)} = f(x, y, y', \ldots, y^{(n-1)}),\quad x \in [a,b]$$

with

$$U_m[y] = \sum_{k=0}^{n-1} (a_{m,k}\, y^{(k)}(a) + b_{m,k}\, y^{(k)}(b)) = g_m$$

where m=1,2,…,n and u_m represent two-points boundary conditions.

Many researchers used (LTPBVP) in many subjects such as Alan [2] used Numerov method to present the solution of this boundary value problem, while Yuji and Weigao [10] discussed the

*Applied Sciences Department, University of Technology/Baghdad

https://doi.org/10.30684/etj.27.11.2
2412-0758/University of Technology-Iraq, Baghdad, Iraq

solvability of this two-points boundary value problem for fourth-order.

In This paper the partition method with the aid of B-Spline functions will be used to approximate a solution of this linear two-points boundary value problem.

2. B-Spline Polynomials:- [4]

B-Spline are the standard representation of smooth non-linear geometry in numerical calculation Schoendery first introduction the B-Spline in 1949, he defined the basis functions using integral convolution B-Spline means spline basis and letter B in B-Spline stands for basis[4], Atef[3] using cubic B-Spline basis function to present optional trajectory planning of manipulators. Higher degree basis functions are given by convolution of multiple basis functions of one degree lower.

In the mathematical subfield of numerical analysis a B-Spline is a spline function which has minimal support with respect to a given degree, smoothness and domain partition. A fundamental theorem states that every spline function of a given degree, smoothness and domain partition can be represented as a linear combination of B-Spline of that same partition. The term B-Spline was coined by Isaac Jacob Schonberg short for basis spline.

The B-Spline can be defined:-

$$B_{k,n} = \binom{n}{k} x^k (1-x)^{n-k} \quad ...(1)$$

where k=0,1,2,…,n and $n \in N$

There are $n+1^{th}$ degree B-Spline polynomials for mathematical convenience, we usually set $B_{k,n}=0$ if $k<0$ or $k>n$.

3. The Main Properties of B-Spline:-

Some properties of B-Spline polynomials are given throughout the following subsections:-

3.1. Converting from the B-Spline Basis to Power Basis:- [11]

Since the power basis $[1,x,x^2,\dots,x^n]$ for the space of polynomials of degree less than or equal to n any B-Spline polynomials of degree n can be written in terms of the power basis. This can be directly calculated using the definition of the B-Spline polynomials and the binomial theorem, as follows:-

$$B_{k,n} = \binom{n}{k} x^k (1-x)^{n-k}$$

$$= \binom{n}{k} x^k \sum_{i=0}^{n-k} (-1)^i \binom{n-k}{i} x^i = \sum_{i=0}^{n-k} (-1)^i \binom{n}{k}\binom{n-k}{i} x^{i+k}$$

$$= \sum_{i=k}^{n} (-1)^{i-k} \binom{n}{k}\binom{n-k}{i-k} x^i$$

$$= \sum_{i=k}^{n} (-1)^{i-k} \binom{n}{i}\binom{i}{k} x^i$$

3.2. Derivatives:- [11]

Derivatives of the n^{th} degree B-Spline polynomial are polynomials of degree n-1.Using the definition of the B-Spline polynomial we can show that this derivative can be written as a linear combination of B-Spline polynomial.

In particular

$$\frac{d}{dx}B_{k,n}(x)=n(B_{k-1,n-1}(x)-B_{k,n-1}(x))$$

for $0 \le k \le n$ …(2)

that the derivative of B-Spline polynomial can be expressed as the product of the degree of the polynomial, multiplied by the difference of the two B-Spline polynomial of degree n-1.

3.3. The B-Spline Polynomials as Basis:- [11]

The B-Spline polynomials of order n form a basis for the space of polynomials of degree less than or equal to n because they span the space of polynomials [any polynomial of degree less than or equal to n can be written as a linear combination of the B-Spline polynomials and they are linearly indepent] that is if there exist constants $k_1,k_2,\ldots,k_n$ that the identity

$$k_0B_{O,n}(x)+k_1B_{i,n}(x)+\ldots+k_nB_{n,n}(x)=0$$

holds for all x, then all the $k_i's$ must be zero.

3.4. A Matrix Representation for B-Spline Polynomials:- [6]

In many applications a matrix formulation for the B-Spline polynomials is useful. Those are straightforward to develop if one only looks at a linear combination of the B-Spline basis functions.

$$B(x)=k_0B_{O,n}(x)+k_1B_{i,n}(x)+\ldots+k_nB_{n,n}(x)$$

It is easy to write this as a dot product of two vectors

$$B(x)=\begin{bmatrix}B_{0,n}(x) & B_{1,n}(x)\ldots B_{n,n}(x)\end{bmatrix}\begin{bmatrix}k_0\\ k_1\\ \vdots\\ k_n\end{bmatrix}$$

which can be converted to:-

$$B(x)=\begin{bmatrix}1 & x & x^2 & \ldots & x^n\end{bmatrix}\begin{bmatrix}b_{0,0} & 0 & 0 & \ldots & 0\\ b_{1,0} & b_{1,1} & 0 & \ldots & 0\\ b_{2,0} & b_{2,1} & b_{2,2} & \ldots & 0\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ b_{n,n} & b_{n,1} & b_{n,2} & \ldots & b_{n,n}\end{bmatrix}$$

Where

$$b_{i,j}=\sum_{i=k}^{n}(-1)^{i-k}*\binom{n}{i}*\binom{i}{k}$$

3.5. Initial and Final Values:- [6]

The initial values of $B_{k,n}$, $0 \le x \le 1$ for n=0,1,2,… are:-

$B_{k,n}(0)$ =0

$B_{0,n}(0)$ =1

$$\dot{B}_{0,n}(0)=-n$$

k=1,2,…,n

$$\dot{B}_{1,n}(0)=n$$

and the final values are

$B_{k,n}(1)$ =0

$B_{n,n}(1)=1$

$k=0,1,\ldots,n-1$

3.6. Differentiation Property:-

Interesting formulas concerning the derivatives of B-Spline polynomials have been derived and given by the following lemma.

Lemma (1):- [6]

The m[th] derivative of B-Spline polynomials $B_{k,n}(x)$ is given by:-

$$B_{k,n}^{(m)} = \frac{n!}{(n-m)!}\sum_{i=0}^{m}(-1)^{i}\binom{m}{i}B_{k+i-m,n-m}(x)$$

Lemma (2):- [6]

the first derivative of B-spline of degree n in terms of B-spline of same degree n is given by the following formula:-

$$\dot{B}_{k,n}(t) = (n-k+1)B_{k-1,n}(x) - (n-2k)B_{k,n}(x) - (k+1)B_{k+1,n}(t) \quad \ldots(3)$$

3.7.Differentiation Operational Matrices for B-Spline Polynomials (D_B):- [6]

If a function y(x) can be approximated by using B-Spline series of length N as follows:-

$$y(x) = \sum_{k=0}^{n} a_k B_{k,n}(x)$$

or in a vector form

$$y(x)=[a_0 \quad a_1 \quad a_3 \ldots a_n]B(x)$$

which can be represented as

$$\dot{y}(x) = [a_0 \quad a_1 \quad a_2 \ \cdots \ a_n]\dot{B}(x)$$

using the derivative of $B_{k,n}(x)$ and eq.(3), we obtain:-

$\dot{B}_{0,n}(x) = -nB_{0,n}(x) - B_{1,n}(x)$

$\dot{B}_{1,n}(x) = nB_{0,n}(x) - (n-2)B_{1,n}(x) - 2B_{2,n}(x)$

$\dot{B}_{2,n}(x) = (n-1)B_{1,n}(x) - (n-4)B_{2,n}(x) - 3B_{3,n}(x)$

$\vdots$

$\dot{B}_{n,n}(x) = B_{n-1,n}(x) - (n-2n)B_{n,n}(x)$

which can also be written in matrix form as:-

$$\begin{bmatrix}\dot{B}_{0,n}(x)\\ \dot{B}_{1,n}(x)\\ \dot{B}_{2,n}(x)\\ \vdots\\ \dot{B}_{n,n}(x)\end{bmatrix} = \begin{bmatrix} -n & -1 & 0 & 0 & 0 & 0\\ n & -(n-2) & -2 & 0 & \vdots & 0\\ 0 & (n-1) & -(n-4) & -3 & \vdots & 0\\ 0 & 0 & (n-k+1) & -(n-2k) & \vdots & 0\\ 0 & \cdots & \cdots & \cdots & 1 & -(n-2n)\end{bmatrix} * \begin{bmatrix}B_{0,n}(x)\\ B_{1,n}(x)\\ B_{2,n}(x)\\ \vdots\\ B_{n,n}(x)\end{bmatrix}$$

or

$$\dot{B}(x) = D_B B(x)$$

where the matrix D_B is the a N×N =(n+1)×(n+1) matrix and it is called B-Spline polynomials differentiation operational matrix.

4. Partition Method:- [7]

Partition method with some kind of basis function has been considered by many researchers, Kareem [8] used this method to solve a higher-order linear Voltera integro-differential equation.

We present the following methods by considering the following boundary value problem:-

$$L[y(x)] = g(x) \qquad x \in D \quad \ldots(4)$$
$$B_L[y] = g_L \qquad x \in \partial D \quad \ldots(5)$$

were L denotes a general differential operator involving special derivatives of y, B_L represents the appropriate boundary conditions and D is the domain with boundary ∂D.

The problem of finding an approximate solution of the boundary value problem (4) is often obtained by assuming an approximation to the solution y(x) by the form:-

$$y_N(x) = \sum_{i=0}^{N} k_i B_{i,n}(x) \ldots(6)$$

where n is a finite positive integer this approximation must satisfy the boundary conditions (5), substituting (6) in (4) we get the residue in the differential equation:-

$$R_N(x) = L[y_N(x)] - g(x) \quad \ldots(7)$$

the residue $R_N(x)$ depends on x as well as on the way that the parameters k_i are chosen. It is obvious that when $R_N(x)=0$, then the exact solution is obtained which is minimize $R_N(x)$ is some sense.

In the partition method, the unknown parameters a_i are chosen to minimize the residue $R_N(x)$ by setting its weighted integral equal to zero i.e $\int_D w_j R(x)dx, \quad J = 0,1,\ldots,N \quad \ldots(8)$

where w_j is prescribed weighting function.

We divide the domain D into N non overlapping sub-domain D_j, j=1,2,…,N. if the weighting function are chosen as follows:-

$$w_j = \begin{cases} 1 & x \in D_j \qquad j = 1,\ldots,N \\ 0 & x \notin D_j \end{cases}$$

hence eq. (4) is satisfied in each of N sub-domain D_j, therefore eq. (8) becomes:-

$$\int_{D_j} R_N(x)dx = 0 \quad , \; j = 1,2,..N \quad \ldots(9)$$

Notice that as much as the sub-domain becomes smaller the residue functions approach zero

i.e

$$Y_N(x) \to y(x) \qquad as \quad N \to \infty$$

5. Partition Method for Solving BVP:-

Suppose we want a numerical solution for linear two-points BVP writable in the form:-

$$\frac{d^n y}{dx^n} + p(x)\frac{d^{n-1}y}{dx^{n-1}} + \mathbf{K} + Q(x)y = g(x) \quad \mathbf{K}(10)$$

subject to the boundary conditions:-

$$y(a)=a \quad , \quad y(b)=b$$

the unknown function y(x) is approximated using B-Spline polynomials:-

$$y_N(x) = \sum_{i=0}^{N} k_i B_{i,N}(X) \quad \mathbf{K}(11)$$

since these approximation must satisfy the boundary condition then get:-

$$y(a)=k_0B_{0,N}(a)+k_1B_{1,N}(a)+k_2B_{2,N}(a) +\mathbf{K}+K_iB_{i,N}(a)=a$$

$$hence \quad k_0=\frac{a-\sum_{i=1}^{N}k_iB_{i,N}(a)}{B_{0,N}(a)}$$

$$y(b)=k_0B_{0,N}(b)+k_1B_{1,N}(b)+k_2B_{2,N}(b) +\mathbf{K}+k_NB_{N,N}(b)=b$$

$$hence \quad k_1=\frac{1}{B_{1,N}(b)}\left[b-B_{0,N}(B)*\left(\frac{a-\sum_{i=1}^{N}k_iB_{i,N}(a)}{B_{0,N}(a)}\right)-\sum_{i=2}^{N}k_iB_{i,N}(b)\right]$$

the technique will be completed as:-

$$y_N(x) = \frac{1}{B_{0,N}(a)}\left(a - \sum_{i=1}^{N} k_i B_{i,N}(a)\right)B_{0,N}(x) + \frac{1}{B_{1,N}(b)}\left[B - \left(\frac{a - \sum_{i=1}^{N} k_i B_{i,N}(a)}{B_{0,N}(a)}\right) * B_{0,N}(b) - \sum_{i=2}^{N} k_i B_{i,N}(b)\right]$$

$$* B_{1,N}(x) + \sum_{i=2}^{N} k_i B_{i,N}(x)$$

using lemma (1) to calculate the derivatives of B-Spline polynomials and using operator form to get:-

$$L[y]=g$$

where the operator L is defined as:-

$$L[y] = \frac{d^n}{dx^n} y_N + p(x)\frac{d^{n-1}}{dx^{n-1}} y_N(x) + \mathbf{K} + Q(x)y_N(x)$$

then the residue equation becomes:-

$$R_N(x) = L[y] - g(x)$$

then by dividing the domain D to subdomian D_J, J=1,2,…,p and assume the integration of residual $R_N(x)$ over the subinterval $[x_{j-1},x_j]$, j=1,2,…,N are vanishes.

i.e:-

$$\int_{x_{j-1}}^{x_j} R_N(x) = 0 \quad , \quad j = 1,2,\mathbf{K},N \quad \mathbf{K}(13)$$

hence (13) can be seen as a system of N-1 equations in N-1 unknown k_m, m=2,…,N-1 this system of non-homogeneous equations can be easily be written in matrix

$$AC=G \quad \mathbf{K}(12)$$

formal:-

Gaussian elimination procedure can be used to determine the solution of values k_m's which satisfy (12) then the approximated solution of eq.(11) will be given.

6. Numerical Examples:-

Example (1):-

Consider the following 2nd order linear two points boundary value problem:-

$$y''(x) = 4y(x)$$

with boundary conditions

y (0) =1.1752 , y (1) =10.0179

while the exact solution is:-

y(x) =sinh(2x+1)

This problem with N=4 using partition by assuming the approximated solution

$$y_4(x) = \sum_{m=0}^{4} k_m B_{m,4}(x)$$

Table (1) presents comparison between the exact and approximated solution which depends on least square error.

Example (2):-

Consider the following 3rd order linear two points boundary value problem:-

$$y'''(x) - y'(x) = \exp(x)$$

With boundary conditions

$$y(0) = 0,\ y(1) = 1,\ y'(1) = 0$$

wile the exact solution is:-

$$y(x) = 4.8618 - 1.4603 * \exp(x)$$

$$- 3.4015 * \exp(-x) + \frac{1}{2} x * \exp(x)$$

This problem is solved with N=4 using partition by assuming the approximate solution

$$y_4(x) = \sum_{m=0}^{4} k_m B_{m,4}(x)$$

Table(2) presents comparison between the exact and approximated solution which depends on least square error.

7. Conclusions:-

Partition method using B-Spline function has been presented for solving linear two points boundary value problem. It has been shown that the proposed method is comparable in accuracy with other method. The results show a marked improvement in the least square errors form which we conclude that the B-Spline functions give better accuracy and more stable than spline functions

8. Reference:-

[1]. “Weighted Residual Methods for Solving ODE and PDE” http:/www.cs.ut.ec/~toomas_1/linalg/dv_2/node8.html 2002

[2]. Alan,L.N. ”Asymptotic Correction of Numerove Eigenvalue Estimates with General Boundary Conditions”, Anzian J.44 (E) PPCI-C19,2003.

[3]. Atef,A.A “Optimal Trajectory Planning of Manipulators” Journal of engineering science and technology.

[4]. B-Spline Curves, internet, lesson (15), Nov.16, 2001, http://www.designer. Mech. Yzu. Edu. Tw/.

[5]. Fox,L.M "The Numerical Solution of Two-Points Boundary Problems in Ordinary Differential Equations" Oxford at the Clarendon prees, 1957.

[6]. Hassen,S.S "Numerical Solution and Algorithms for Optimal Control Problems" M.Sc, thesis, University of Technology 2006.

[7]. Jain,M.K "Numerical Solution of Differential Equations" Department of Mathematics, India Institute of Technology, 1979.

[8]. kareem,R.S "Approximated Treatment of Higher-order Linear Volterra-integro Differential Equations" M.Sc, thesis, University of Technology 2003.

[9]. Milan,k & Vladimir,A "Numrical Solution of Nonlinear Boundary Value Problems with Applications" Prentic-Hall, INC1983.

[10]. Yuji,L.& Weigao,G. "Solvability of Two-points Boundary Value Problems for Fourth-order Nonlinear Differential Equations at Resonance " Journal from an analysis and its applications volume 22 (2003), No 4, 977-989.

[11] . Kennth.I .J "Bernstein polynomials" visualization and graphics research group department of computer science university of California, Davis.

Table (1)

x	The Exact solution	The approximated solution
0	1.1752	1.1752
0.1	1.5095	1.5109
0.2	1.9043	1.9082
0.3	2.3756	2.3801
0.4	2.9422	2.9452
0.5	3.6269	3.6274
0.6	4.4571	4.4562
0.7	5.4662	5.4666
0.8	6.6947	6.6988
0.9	8.1919	8.1988
1	10.0179	10.0179
L.S.E		0.00001

Table (2)

x	The Exact solution	The approximated solution
0	0	0
0.1	0.2254	0.2264
0.2	0.4154	0.4171
0.3	0.5732	0.5752
0.4	0.7015	0.7034
0.5	0.8032	0.8047
0.6	0.8808	0.8817
0.7	0.9367	0.9371
0.8	0.9736	0.9737
0.9	0.9938	0.9938
1	1	1
L.S.E		0.00001