Expansion Method For Solving Linear Delay Integro-Differential Equation Using B- Spline Functions

Hayat Adel Ali*

Accepted on:2/12/2008
Received on:5/3/2009

Abstract

The main goal of this paper lies briefly in submitting and modifying some methods for solving linear delay integro-differential equations (L-DIDEs) containing three types (retarded, neutral and mixed) numerically by employing expansion method (collocation and partition) with the aid of B-spline polynomials as basis functions to compute the numerical solutions of (L-DIDEs). Three numerical examples are given for determining the results of this method.

Keywords: Linear Delay Integro- Differential Equation (L-DIDE), B-Spline Functions.

طريقة التوسيع لحل المعادلات التكاملية – التفاضلية التباطؤية الخطية

باستخدام الدوال التوصيلية

1. Introduction

Delay integro-differential equations (DIDEs) are equations having delay argument. They arise in many realistic models of problems in science, engineering and medicine [6]. Only in the last few years has much effort in behavior of solution of delay differential and delay integro- differential equations, i.e. equations in which the highest order derivative of unknown function appear with delay which is called neutral delay integro- differential equation as well as retarded differential equation.

The general form of linear delay integro- differential equation is:

\[ f(t - \tau) = g(t) + \int k(t, s)f(s - \tau)ds \ldots (1) \]

where \( f(t) \) is the unknown function and \( g(t), k(t,s) \) are known continuous functions. Eq. (1) is

*Application Science Department, University of Technology/ Baghdad
classified as Volterra delay integro- differential equation if \( b(t)=t \), otherwise when \( b(t)=b \) where \( b \) is constant it’s called Fredholm delay integro- differential equation.

Three types of eq. (1) can be introduced depending on \( (\tau_1, \tau_2) \) as follows:-

(1) Retarded integro- differential equation if \( \tau_1=0 \) and \( \tau_2>0 \)

\[
f(t) = g(t) + \int_{t_0}^{t} k(t,s)f(s-\tau_2)ds \quad (2)
\]

(2) Neutral integro- differential equation if \( \tau_1>0 \) and \( \tau_2=0 \)

\[
f(t-\tau_1) = g(t) + \int_{t_0}^{t} k(t,s)f(s)ds \quad (3)
\]

(3) Mixed integro- differential equation when \( \tau_1>0 \) and \( \tau_2>0 \)

\[
f(t-\tau_1) = g(t) + \int_{t_0}^{t} k(t,s)f(s-\tau_2)ds \quad (4)
\]

Many researchers used linear delay integro- differential equation in some subjects such as Hopkins, T [1] found the numerical solution of stochastic DIDEs in population dynamics as well as Xuyang lou [14] studied delay integro differential equation, modeling neural field.

2. B- Spline Polynomials and Properties:-

B-splines are the standard representation of smooth non linear geometry in numerical calculation. Schoendery [12] first introduced the B-spline in 1949. He defines the basis functions using integral convolution. B-spline means spline basis and letter B in B-spline stands for basis.

**Definition (1)**

Given m+1 knots \( t_i \) in \([0, 1]\) with \( t_0 < t_1 < t_2 < \ldots < t_m \)

A B-spline of degree \( n \) is a parametric curve

\[
B: [0, 1] \rightarrow \mathbb{R}^2
\]

Composed of basis B-spline of degree \( n \)

\[
B(t) = \sum_{i=0}^{m} p_i B_i(t), \quad t \in [0,1]
\]

The \( p_i, i=0, 1, \ldots, m+1 \) are called control points or anchor points or de Boor point. A polygon can be constructed by connecting the de Boor points with lines starting with \( p_0 \) and finishing with \( p_n \) this polygon is called the de Boor polygon.

The \( m-n \) basis B-spline of degree \( n \) can be defined using the cox-de Boor recursion formula.

\[
B_{i,n}(t) = \begin{cases} 
1 & \text{if } t_i \leq t \leq t_{i+1} \\
0 & \text{otherwise} 
\end{cases} 
\]

\[
B_{i,n}(t) = \frac{t-t_{i-1}}{t_{i+n}-t_{i-1}} B_{i-1,n}(t) + \frac{t_{i+n}-t}{t_{i+n}-t_{i-1}} B_{i+1,n}(t) \quad L (5)
\]

when the knots are equidistant we say the B-spline is uniform otherwise we call it non- uniform.

The B-spline can be defined in another way like:-

\[
B_{i,n}(t) = \binom{n}{k} (1-t)^{n-k}, \quad \text{for } i=0, 1, \ldots, n
\]
where
\[
\binom{n}{k} = \frac{n!}{k!(n-k)!}
\]

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

Some important types of B-spline polynomials which are used in this work are introduced as follows:-

2.1 Constant B-spline \(B_{k,0}(t)\)

The constant B-spline is the simplest spline. It is defined on only one knot span and is not even continuous on the knots. It is a just indicator function for the different knot spans.

\[
B_{k,0}(t) = \begin{cases} 
1 & \text{if } t_k \leq t \leq t_{k+1} \\
0 & \text{otherwise}
\end{cases}
\]

2.2 Linear B-spline \(B_{k,1}(t)\)

The linear B-spline is defined on two consecutive knot spans and is continuous on the knots, but not differentiable.

\[
B_{k,1}(t) = \begin{cases} 
\frac{t-t_k}{t_{k+1}-t_k} & \text{if } t_k \leq t \leq t_{k+1} \\
\frac{t_{k+1}-t}{t_{k+1}-t_{k+2}} & \text{if } t_{k+2} \leq t \leq t_{k+1} \\
0 & \text{otherwise}
\end{cases}
\]

\(0r\ B_{0,1}(t) = 1-t, B_{1,1}(t) = t\)

2.3 Quadratic B-spline \(B_{k,2}(t)\)

Quadratic B-spline with uniform knot-vector is a commonly used form of B-spline. The blending function can easily be recalculated, and is equal to each segment in this case:-

\[
B_{k,2}(t) = \begin{cases} 
\frac{1}{2}t_k^2 - t_k^3 + t_k & \text{if } t_k \leq t \leq t_{k+1} \\
\frac{1}{2}(t-t_k)^2 & \text{if } t_{k+1} \leq t \leq t_{k+2} \\
\frac{1}{2}(t-t_{k+2})^2 & \text{if } t_{k+2} \leq t \leq t_{k+3} \\
0 & \text{otherwise}
\end{cases}
\]

or \(B_{2,2}(t) = t^2, B_{1,2}(t) = 2t(1-t)\),

\(B_{0,2}(t) = (1-t)^2\)

2.4 Cubic B-spline \(B_{k,3}(t)\)

Cubic B-spline with uniform knot-vector is the most commonly used form of B-spline. The blending function can easily be recalculated and is equal to each segment in this case:-

\[
B_{3,3}(t) = t^3, B_{2,3}(t) = 3t^2(1-t), B_{1,3}(t) = 3t^2(1-t)
\]

Some important properties of B-spline polynomials are given as follows:-

**Property (1):**

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 polynomials [12].

In particular

\[
\frac{d}{dt}B_{k,n}(t) = nB_{k,n-1}(t) - B_{k+1,n}(t)
\]

That derivative of a B-spline polynomial can be expressed as the degree of the polynomial, multiplied
by the difference of two B-spline polynomials of degree n-1.

**Property (2):**

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 of degree less than or equal to n can be written as a linear combination of the B-spline polynomials and they are linearly independent that is if there exist constants $c_0$, $c_1$, ..., $c_n$ that the identity

$$0=c_0B_{0,n}(t)+c_1B_{1,n}(t)+...+c_nB_{n,n}(t)$$

holds for all t, then all the $c_i$'s must be zero.

3. Expansion Method:

This method illustrates important approaches coming from the field of approximation theory, in which the unknown solution $f(t)$ is expanded in terms of a set of known functions (B-spline polynomials) as follows

$$f(t) \equiv f_n(t) = \sum_{i=0}^{n} c_i B_{i,n}(t)$$

The unknown then being the expansion coefficients $c_i$. An algorithm based on the above approximation is an expansion method in this work collocation and partition methods are considered as expansion methods.

3.1 Solution of L-DIDEs Using Collocation Method with the Aid of B-spline:

First consider the linear retarded integro-differential equation of the form

$$\frac{df(t)}{dt} = g(t) + \int_{a}^{b} k(t,s)f(s-\tau)ds \quad L \hspace{1cm} (7)$$

By approximating the unknown function $f(t)$ using B-spline polynomial as a basis function we have

$$f(t) \equiv f_n(t) = \sum_{j=0}^{n} c_j B_{j,n}(t) \quad for \quad j=0,1,...,N \quad L \hspace{1cm} (8)$$

Substituting eq.(8) into eq.(7) for $f(t)$ and with $t=t_j$ and by using property (1) for $f'(t)$ we get the following formula

$$\sum_{i=0}^{n} \int_{a}^{b} k(t,s)B_{i,n}(s-\tau)ds = g(t) + \sum_{j=0}^{n} c_j B_{j,n}(s-\tau)$$

So

$$\sum_{i=0}^{n} \int_{a}^{b} k(t,s)B_{i,n}(s-\tau)ds = g(t) + \sum_{j=0}^{n} c_j B_{j,n}(s-\tau)$$

for $j=0,1,...,N$

then

$$\sum_{i=0}^{n} \int_{a}^{b} k(t,s)B_{i,n}(s-\tau)ds = g(t) + \sum_{j=0}^{n} c_j B_{j,n}(s-\tau)$$

Second consider the linear neutral integro-differential equation of the form

$$\frac{df(t)}{dt} = g(t) + \int_{a}^{b} k(t,s)f(s)ds \quad L \hspace{1cm} (10)$$

Substituting eq.(8) into eq.(10) we get the equation

$$\sum_{i=0}^{n} \int_{a}^{b} k(t,s)B_{i,n}(s-\tau)ds = g(t) + \sum_{j=0}^{n} c_j B_{j,n}(s-\tau)$$

then

2010

PDF created with pdfFactory Pro trial version www.pdffactory.com
Expansion Method For Solving Linear Delay Integro-Differential Equation Using B-Spline Functions

\[
\sum \{NB_{i\tau}(t, \tau_{-}) - B_{i\tau}(t, \tau_{-})\} \int [0, s] B_{i\tau}(s) dA = g(t)
\]

Third consider the L-MIDE's of the form

\[
\frac{df(t)}{dt} = g(t) + \int [k(t, s) f(s - \tau_{-})] ds \tag{12}
\]

By Substituting eq. (8) into eq. (12) and follows the previous steps we have the following equation as a results:-

\[
\sum \{NB_{i\tau}(t, \tau_{-}) - B_{i\tau}(t, \tau_{-})\} = \sum \int B_{i\tau}(s - \tau_{-}) k_{i\tau}(s) ds = g(t)
\]

\[
\ldots(13)
\]

for \( j=0,1,...,N \)

These equations involve \( n+1 \) unknown coefficients \( [c_{i}] \), then we may select \( n-1 \) points \( \{t_{1}, t_{2}, ..., t_{N-1}\} \) in the range of integration and require \( f_{0}(t) \) to satisfy the delay integro – differential equation at just these \( n-1 \) points. This method requires us to solve just the system of \( n+1 \) linear equations for each type of delay integro differential equations (retarded, neutral and mixed), these system of \( n+1 \) Linear equations for the coefficients \( c_{i} \) can be written in matrix form as follows:-

\[
AC = G
\]

where

\[
A = \begin{bmatrix}
\alpha_{0} & \alpha_{0} & L & \alpha_{0} \\
\alpha_{0} & \alpha_{0} & L & \alpha_{0} \\
\alpha_{0} & \alpha_{0} & L & \alpha_{0} \\
M & M & O & M
\end{bmatrix}
\]

\[
A_{i} = \{NB_{i\tau}(t, \tau_{-}) - B_{i\tau}(t, \tau_{-})\} \int [0, s] B_{i\tau}(s) dA = g(t)
\]

\[
\ldots(14)
\]

k=1,2

the existence of \( (\tau_{k}) \) depends on the type of the equation (retarded, neutral and mixed)

\[
C = [c] , \quad G = [g(t)]
\]

By solving this system by Gauss elimination procedure to find \( c_{i} \)’s for \( i=0,1,...,N \).

3.2 Solution of L-DIDEs Using Partition Method with the Aid of B-spline:-

In this method we divide the domain \( R \) into \( P \) non over lapping sub domains \( R_{j} \) \( j=1,2,...,p \), if the weighting functions are chosen as follows

\[
w_{j} = \begin{cases} 
1 & \text{if } t \in R_{j} \\
0 & \text{if } t \notin R_{j}
\end{cases}
\]

then the delay integro – differential equation is satisfied on the average in each of the \( P \) sub domains \( R_{j} \), the required equation for partition method become

\[
\int_{R_{j}} E(t) dt = 0 \quad j=0,1,...,N \quad \ldots(15)
\]

where \( E(t) \) is called residue equation and hopefully it approaches zero on \( R_{j} \).

or which we have the residue equation for (retarded, neutral and mixed) integro-differential equation respectively as follows:-

for retarded equation we have
Expansion Method For Solving Linear Delay
Integro-Differential Equation Using B-Spline Functions

\[ E(t) = \sum_{i=0}^{N+1} c_i \int_0^t k(t, s) B_i(s-t) r(s) ds \]

for neutral equation we get

\[ E(t) = \sum_{i=0}^{N+1} c_i \int_0^t k(t, s) B_i(s-t) r(s) ds \]

for mixed equation we get

\[ E(t) = \sum_{i=0}^{N+1} c_i \int_0^t k(t, s) B_i(s-t) r(s) ds \]

Substituting these equations into eq.(15) we get:

\[ \sum c_i \int_0^t k(t, s) B_i(s-t) r(s) ds = g(t) \]

These equations give system of (N+1) Linear equations in N+1 unknown coefficients \( c_i \), \( i=0,1,...,N \). Rewriting the above equations in a matrix form as follows:-

\[ \mathbf{MC} = \mathbf{G} \]

where

\[ \mathbf{M} = \begin{bmatrix} \beta_0 & \beta_1 & \beta_2 & \cdots & \beta_N \\ \mathbf{M} & \mathbf{M} & \mathbf{M} & \cdots & \mathbf{M} \\ \beta_0 & \beta_1 & \beta_2 & \cdots & \beta_N \\ \end{bmatrix} \]

by using Gauss elimination procedure we find the values of \( c_i \)'s.

4. Numerical Examples:-

Example (1): Consider the following retarded Volterra integro-differential equations:

\[ f(t) = (1-t^2) + \int_0^t t s f(s-1) ds \quad 0 \leq t \leq 1 \]

with exact solution \( f(t) = t + 1 \)

Assume the approximate solution is:

\[ f(t) = c_0 (1-t) + c_1 t \]

This problem is solved using collection method and partition method with the aid of B-spline functions as basis functions, the solution of \( f(t) \) is obtained as shown in table (1), with \( N=10, h=0.1 \) and \( t_0=ih, i=0,1,...,N \).

Example (2): Consider the following neutral integro-differential equations:

\[ f'(t) = 4(t-1)^3 - \frac{t^4}{5} + \int_0^t f(s) ds \quad 0 \leq t \leq 1 \]

with exact solution \( f(t) = t^4 \)

Assume the approximate solution is:

\[ f(t) = \sum c_i B_i(t) \]

\[ f(t) = c_0 (1-t)^3 + 4c_1 (1-t)^2 + 6c_2 (1-t) + 4c_3 (1-t) + c_4 t \]

Table (2) lists the results obtained by achieving collection method and partition method with the aid of B-spline function.

Example (3):
Consider the following mixed Fredholm integro-differential equations:

\[ f'(t-1) = 2t + \frac{25}{12} \int_0^1 f(t-1/2)dt \quad 0 \leq t \leq 1 \]

with exact solution \( f(t) = t' \)

Assume the approximate solution is:

\[ f(t) = \sum c_i B_i(t) \]

The results of this example list in table (3) which obtained by using collection and partition method with the aid of B-spline.

**Conclusions**

1. The expansion method including (collocation and partition) methods with the aid of B-spline polynomials as a basis function which are used in this paper have proved their effectiveness in solving (L-DIDEs) numerically and finding accurate results.

2. B-spline function depends on N as N increased, the error term is decreased.

3. The results show a marked improvement in the least square errors from which we conclude that.

**References**


[10] Al- Faour Dr. Omar M.; Solving a class of system of Volterra Integro - Differential Equations Using B-spline function, University of Keikouk.


[14] Xuyang lou (1), Baotong cui (1). Boundedness and stability for integrodifferential equations modeling neural field with time delay.

Table (1)

<table>
<thead>
<tr>
<th>ti=ih</th>
<th>Exact Solution</th>
<th>Collection with(B-spline)</th>
<th>Partition with(B-spline)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>0.1</td>
<td>1.1000</td>
<td>1.1000</td>
<td>1.1000</td>
</tr>
<tr>
<td>0.2</td>
<td>1.2000</td>
<td>1.2000</td>
<td>1.2000</td>
</tr>
<tr>
<td>0.3</td>
<td>1.3000</td>
<td>1.3000</td>
<td>1.3000</td>
</tr>
<tr>
<td>0.4</td>
<td>1.4000</td>
<td>1.4000</td>
<td>1.4000</td>
</tr>
<tr>
<td>0.5</td>
<td>1.5000</td>
<td>1.5000</td>
<td>1.5000</td>
</tr>
<tr>
<td>0.6</td>
<td>1.6000</td>
<td>1.6000</td>
<td>1.6000</td>
</tr>
<tr>
<td>0.7</td>
<td>1.7000</td>
<td>1.7000</td>
<td>1.7000</td>
</tr>
<tr>
<td>0.8</td>
<td>1.8000</td>
<td>1.8000</td>
<td>1.8000</td>
</tr>
<tr>
<td>0.9</td>
<td>1.9000</td>
<td>1.9000</td>
<td>1.9000</td>
</tr>
<tr>
<td>1</td>
<td>2.0000</td>
<td>2.0000</td>
<td>2.0000</td>
</tr>
<tr>
<td>L.S.E</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>t</td>
<td>Exact solution</td>
<td>Collocation with (B-spline)</td>
<td>Partition with (B-pline)</td>
</tr>
<tr>
<td>----</td>
<td>----------------</td>
<td>-----------------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.1</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>0.2</td>
<td>0.0016</td>
<td>0.0016</td>
<td>0.0016</td>
</tr>
<tr>
<td>0.3</td>
<td>0.0081</td>
<td>0.0081</td>
<td>0.0081</td>
</tr>
<tr>
<td>0.4</td>
<td>0.0256</td>
<td>0.0256</td>
<td>0.0256</td>
</tr>
<tr>
<td>0.5</td>
<td>0.0625</td>
<td>0.0625</td>
<td>0.0625</td>
</tr>
<tr>
<td>0.6</td>
<td>0.1296</td>
<td>0.1296</td>
<td>0.1296</td>
</tr>
<tr>
<td>0.7</td>
<td>0.2401</td>
<td>0.2401</td>
<td>0.2401</td>
</tr>
<tr>
<td>0.8</td>
<td>0.4096</td>
<td>0.4096</td>
<td>0.4096</td>
</tr>
<tr>
<td>0.9</td>
<td>0.6561</td>
<td>0.6561</td>
<td>0.6561</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>L.S.E</td>
<td>0.0000</td>
<td>0.0000</td>
<td></td>
</tr>
</tbody>
</table>
### Table (3)

<table>
<thead>
<tr>
<th>t</th>
<th>Exact solution</th>
<th>Collocation with(B-spline)</th>
<th>Partition with(B-spline)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.1</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>0.2</td>
<td>0.04</td>
<td>0.04</td>
<td>0.04</td>
</tr>
<tr>
<td>0.3</td>
<td>0.09</td>
<td>0.09</td>
<td>0.09</td>
</tr>
<tr>
<td>0.4</td>
<td>0.016</td>
<td>0.016</td>
<td>0.016</td>
</tr>
<tr>
<td>0.5</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
</tr>
<tr>
<td>0.6</td>
<td>0.36</td>
<td>0.36</td>
<td>0.36</td>
</tr>
<tr>
<td>0.7</td>
<td>0.49</td>
<td>0.49</td>
<td>0.49</td>
</tr>
<tr>
<td>0.8</td>
<td>0.64</td>
<td>0.64</td>
<td>0.64</td>
</tr>
<tr>
<td>0.9</td>
<td>0.81</td>
<td>0.81</td>
<td>0.81</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>L.S.E</td>
<td>0.0000</td>
<td>0.0000</td>
<td></td>
</tr>
</tbody>
</table>