Solving a Class of System of Volterra Integro-Differential Equations Using $B-spline$ Functions

*Omar M. Al-Faour

Date of acceptance 6/93/2006

Abstract:

In this paper we propose an approach to the approximation for a system of first order linear Volterra integro-ordinary differential equations LVIODEs. In this approach, $B-spline$ basis functions which is dependent on control parameters is first formulated using matrix equations. We then focus on the two features of the $B-spline$ functions, convergence and stability.

Numerical results, including the solution of LVIODEs show the effectiveness that can be achieved using the proposed formula of the $B-spline$ functions.

1. Introduction

A large class of scientific and engineering problems is modeled by integral equation, integro-differential equation, integro-partial differential equation or coupled ordinary and partial differential equations, which can be described as a system of linear VIDEs in Banach space[1].

In this paper, $B-spline$ functions including different orders have been applied to find approximate solution for system of LVIODEs.

The general form for the system of first order LVIODEs is:

$$u_i'(x)+P_i(x)u_i(x)=f_i(x)+\sum_{j=1}^{m}\int_0^x k_{ij}(x,t)u_j(t)dt;\quad i=1,2,\ldots,m \tag{1}$$

where $x\in I=[0,1]$, with the initial conditions

$$u_i(0)=u_i;\quad i=1,2,\ldots,m \tag{2}$$

where the functions f_i and P_i; $i=1,2,\ldots,m$ are assumed to be continuous on I, and k_{ij}; $i,j=1,2,\ldots,m$ denotes given continuous functions.

2. $B-spline$ functions

$B-splines$ are the standard representation of smooth non-linear geometry in numerical calculations. Schoenberg first introduced the $B-spline$ in 1949. He defined the basis functions using integral convolution (the " B " in $B-spline$ stands for "basis")[2],[3],[4].

Let X be a set of non-decreasing numbers,

$\cdots\le x_{-2}\le x_{-1}\le x_0\le x_1\le x_2\le\cdots$.

The x_i's are called knots, the set X is the knot vector and the half open interval $[x_i,x_{i+1})$ is the ith knot span. Then the ith $B-spline$ basis functions of degree k, $B_i^k(x)$, can be defined recursively as follows [5]:

$$B_i^k(x)=\frac{x-x_i}{x_{i+k}-x_i}B_i^{k-1}(x)+\frac{x_{i+k+1}-x}{x_{i+k+1}-x_{i+1}}B_{i+1}^{k-1}(x) \tag{3}$$

where

$$B_i^0(x)=\begin{cases}1 & ,if\ x_i\le x\le x_{i+1}\\ 0 & ,otherwise\end{cases} \tag{4}$$

The above formula is usually referred to as "cox-de Boor recursion formula". The x_i's are called parametric knot

*Dr.\ University of TechnologY/ School of Applied Science

values, for an open curve they are given by:

$$x_i = \begin{cases} 0 & ,i < k+1 \\ i-k & ,k+1 \le i \le n \\ n-k+1 & ,i > n \end{cases}$$

where $0 < i < n+k+1$, and the range of x is $0 \le x \le n-k+1$.

In this paper, we will assume that $n = k$, hence; the parametric knot values x_i will be given by

$$x_i = \begin{cases} 0 & ,i < k+1 \\ i-k & ,k+1 \le i \le k \\ 1 & ,i > k \end{cases} \qquad (5)$$

where $0 \le i \le 2k+1$, and the range of is $0 \le x \le 1$, that is,

$$\begin{bmatrix} x_0 & x_1 & x_2 & \dots & x_k & x_{k+1} & \dots & x_{2k} & x_{2k+1} \end{bmatrix}$$
$$= \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 1 & \dots & 1 & 1 \end{bmatrix}$$

The theory of $B-spline$ curves separates the degree of the resulting curve from the number of the given control points. The $B-spline$ curve is defined by $k+1$ control points b_i. In particular, for any set of points $b_0, b_1, \dots, b_k$ and for any x, the expression,

$$B^k(x) = b_0\, B_0^k(x) + b_1\, B_1^k(x) + \cdots b_k\, B_k^k(x)$$
$$0 \le x \le x_{\max} \qquad (6)$$

is an affine combination of the set of points $b_0, b_1, \dots, b_k$ and if $0 \le x \le 1$, it is a convex combination of the points.

To understand the recursive nature of the $B-spline$ functions, an important formula for kth order $B-spline$ functions will be derived in this section.

- 1^{st} order $B-spline$ $B_i^1(x)$:

This kind of $B-spline$ is called linear spline, it is defined from the recurrence relation defined by eq. (3). Then we have:

$$B_i^1(x) = \begin{cases} \dfrac{x - x_i}{x_{i+1} - x_i} & ,x_i \le x < x_{i+1} \\ \dfrac{x_{i+2} - x}{x_{i+2} - x_{i+1}} & ,x_{i+1} \le x < x_{i+2} \\ 0 & ,otherwise \end{cases}$$

For 1^{st} order $B-spline$, we have $k = 1$, and obtain the following parameters knot x_i; $i = 0,1,2,3$

$$\begin{bmatrix} x_0 & x_1 & x_2 & x_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix} \qquad (7)$$

these values can be put into eq. (6) to get:

$$B^1(x) = b_0 B_0^1(x) + b_1 B_1^1(x), \quad 0 \le x \le 1$$

$$= b_0 \begin{cases} \dfrac{x - x_0}{x_1 - x_0} & ,x_0 \le x < x_1 \\ \dfrac{x_2 - x}{x_2 - x_1} & ,x_1 \le x < x_2 \\ 0 & ,otherwise \end{cases} + \qquad (8)$$
$$b_1 \begin{cases} \dfrac{x - x_1}{x_2 - x_1} & ,x_1 \le x < x_2 \\ \dfrac{x_3 - x}{x_3 - x_2} & ,x_2 \le x < x_3 \\ 0 & ,otherwise \end{cases}$$

Combining eqs. (7) and (8) to obtain

$$B^1(x) = b_0(1-x) + b_1\, x, \quad 0 < x < 1 \quad (9)$$

that is, $B_0^1(x) = 1 - x$ and $B_1^1(x) = x$ which gives us the new formula for the *1^{st} order $B-spline$ functions.*

- 2^{nd} order $B-spline$ $B_i^2(x)$:

This kind of $B-spline$ is called quadratic spline, using eq. (3), we get:

$$B_i^2(x) = \begin{cases} \dfrac{(x-x_i)^2}{(x_{i+2}-x_i)(x_{i+1}-x_i)} & , x_i \le x < x_{i+1} \\ \dfrac{(x-x_i)(x_{i+2}-x)}{(x_{i+2}-x_i)(x_{i+2}-x_{i+1})} + \dfrac{(x_{i+3}-x)(x-x_{i+1})}{(x_{i+3}-x_{i+1})(x_{i+2}-x_{i+1})} & , x_{i+1} \le x < x_{i+2} \\ \dfrac{(x_{i+3}-x)^2}{(x_{i+3}-x_{i+1})(x_{i+3}-x_{i+2})} & , x_{i+2} \le x < x_{i+3} \\ 0 & , otherwise \end{cases}$$

For 2^{nd} order $B-spline$, we have $k=2$, and obtain the following parameters $x_i; i=0,1,2,\ldots,5$

$$\begin{bmatrix} x_0 & x_1 & x_2 & x_3 & x_4 & x_5 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \qquad (10)$$

The formula for the *2^{nd} order $B-spline$ functions* is given by:

$$B^2(x) = b_0(1-x)^2 + 2b_1 x(1-x) + b_2 x^2, \quad 0 \le x \le 1 \qquad (11)$$

that is, $B_0^2(x) = (1-x)^2$, $B_1^2(x) = 2x(1-x)$ and $B_2^2(x) = x^2$.

- 3^{rd} order $B-spline$ $B_i^3(x)$:

In the same way as in 1^{st} and 2^{nd} order $B-spline$ functions, the formula for the *3^{rd} order $B-spline$ functions* can be calculated to be

$$B^3(x) = b_0(1-x)^3 + 3b_1 x(1-x)^2 + 3b_2(1-x)x^2 + b_3 x^3, \quad 0 \le x \le 1 \qquad (12)$$

that is, $B_0^3(x) = (1-x)^3$, $B_1^3(x) = 3x(1-x)^2$, $B_2^3(x) = 3x^2(1-x)$ and $B_3^3(x) = x^3$.

- k^{th} order $B-spline$ $B_i^k(x)$:

From the above discussion concerning the $B-spline$ curves of orders 1, 2, and 3, we can conclude the following formula for the k^{th} order $B-spline$ functions

$$B_i^k(x) = \frac{k!}{(k-i)!\,i!} x^i (1-x)^{k-i}, 0 \le x \le 1 \qquad (13)$$

There are kth degree $B-spline$ polynomials. For mathematical convenience, we usually set $B_i^k(x) = 0$, if $i < 0$ or $i > k$.

Eq. (13) is very essential in our work as well as in other applications.

3. Solution for System of LVIODEs Using $B-spline$ Functions of Different Orders

In this section we shall discuss a variety of different $B-spline$ formulas which holds for LVIODEs.

Recall eqn. (1) and (2)

$$u_i'(x) + P_i(x)u_i(x) = f_i(x) + \sum_{j=1}^{m} \int_0^x k_{ij}(x,t)u_j(t)\,dt; \quad i = 1,2,\ldots,m \qquad (14)$$

where $x \in I = [0,1]$, with the initial conditions

$$u_i(0) = u_i; \quad i = 1,2,\ldots,m \qquad (15)$$

where the functions f_i and P_i; $i = 1,2,\ldots,m$ are assumed to be continuous on I, and k_{ij}; $i,j = 1,2,\ldots,m$ denotes given continuous functions.

Now the basic ideas of using the formulas of $B-spline$ functions derived in the pervious section are presented for approximating LVIODEs.

In this approach, each unknown function $u_i(x)$; $i = 1,2,\ldots,m$ in eq. (14) is approximated by k^{th} order $B-spline$ functions defined on a set of knots $\{0 = x_0^i, x_1^i, \ldots, x_k^i = 1\}$, that is,

$$u_i(x) = b_0^i B_0^k(x) + b_1^i B_1^k(x) + \cdots + b_k^i B_k^k(x); \; i = 1,2,\ldots,m \qquad (16)$$

In other word, $u_i(x)$; $i = 1,2,\ldots,m$, can be uniquely identified with a control parameters vector b^i with $b^i = \{b_0^i, b_1^i, \ldots, b_k^i\}$.

where k is the order of $B-spline$ functions $B_i^k(x)$.

- For the case of linear $B-spline$ basis functions, we take $k=1$ in eq. (16), that is,

$$u_i(x) = b_0^i B_0^1(x) + b_1^i B_1^1(x) \;, 0 \le x \le 1 \qquad (17)$$

with the control parameters b_0^i and b_1^i, $i = 1,2,\ldots,m$, to be found as follows:

put $x=0$ and $x=1$ into eq. (17) to obtain $u_i(0) = b_0^i$ and $u_i(1) = b_1^i$. Hence eq. (17) becomes

$$u_i(x) = u_i(0)B_0^1(x) + u_i(1)B_1^1(x) \qquad (18)$$

Differentiate eq. (18) w.r.t. x, yields

$$u_i'(x) = -u_i(0) + u_i(1) \qquad (19)$$

Substituting (18)-(19) into (14) to obtain

$$-u_{i0} + u_{i1} + P_i(x)\left(u_{i0}\, B_0^1(x) + u_{i1}\, B_1^1(x)\right) = f_i(x) + \sum_{j=1}^{m}\int_0^x k_{ij}(x,t)\left(u_{i0}\, B_0^1(t) + u_{i1}\, B_1^1(t)\right)dt; \qquad (20)$$

where $u_{i0} = u_i(0)$ and $u_{i1} = u_i(1)$; $i = 1,2,\ldots,m$, $0 \le x \le 1$.

Finally, put $x=1$ into eq. (20) and using trapezoidal rule to find the control parameters $u_i(1) = b_1^i$; $i = 1,2,\ldots,m$ then substituting these values in the linear $B-spline$ expansion (18) , we obtain an approximation for the unknown functions $u_i(x)$; $i = 1,2,\ldots,m$.

- To process eq. (14) with quadratic $B-spline$ functions $B_i^2(x)$, take $k=2$ in eq. (13) to obtain:

$$u_i(x) = b_0^i B_0^2(x) + b_1^i B_1^2(x) + b_2^i B_2^2(x), \quad 0 \le x \le 1 \qquad (21)$$

In this case, the control parameters vector $b^i = \{b_0^i, b_1^i, b_2^i\}$ is evaluated as follows:

$b_0^i = u_i(0)$, $b_1^i = \dfrac{1}{2}\dfrac{du_i(0)}{dx} + b_0^i$ and $b_2^i = u_i(1)$.

The control parameters b_0^i are evaluated from the initial conditions (15) while the control parameters b_1^i are evaluated by putting $x=0$ into eq. (14) to get:

$$u_{i0}' = f_{i0} - P_{i0}\, u_{i0} \;; \quad i = 1,2,\ldots,m \qquad (22)$$

where $u_{i0} = u_i(0)$, $f_{i0} = f_i(0)$, $P_{i0} = P_i(0)$ and $u_{i0}' = u_i'(0)$.

The values u_{i0}, P_{i0} and f_{i0} are known , therefore u_{i0}' can be found. After substituting b_0^i, b_1^i and eq. (21) in eq. (14) and take $x=1$, the control parameters b_2^i can be found with the aid of trapezoidal rule. When the values of the control parameters b_0^i, b_1^i and b_2^i are substituted into eq. (21), the solution to $O(x^2)$ is obtained.

- Take $k=3$ in eq. (13) to get an expansion for $u_i(x)$ using third order $B-spline$ functions $B_i^3(x)$, that is:

$$u_i(x) = b_0^i B_0^3(x) + b_1^i B_1^3(x) + b_2^i B_2^3(x) + b_3^i B_3^3(x), \quad 0 \le x \le 1 \qquad (23)$$

were the control parameters vector $b^i = \{b_0^i, b_1^i, b_2^i, b_3^i\}$ can be obtained as follows:

$b_0^i = u_{i0}$, $b_1^i = \dfrac{1}{3}\dfrac{du_i(0)}{dx} + b_0^i$,

$b_2^i = \dfrac{1}{6}\dfrac{d^2u_i(0)}{dx^2} - b_0^i + 2b_1^i$ and $b_3^i = u_i(1)$.

The control parameters b_0^i; $i = 1,2,\ldots,m$, are evaluated using the initial conditions (15), b_1^i; $i = 1,2,\ldots,m$, are computed using eq. (22) while the values of b_1^i; $i = 1,2,\ldots,m$, are obtained after differentiated eq. (14) w.r.t. x then put $x = 0$ to get:

$$u_{i0}'' = f_{i0}' - P_{i0}' u_{i0} - P_{i0} u_{i0}' + \sum_{j=1}^{m} k_{ij}(0,0) u_{j0}$$
$$i = 1,2,\ldots,m \qquad (24)$$

The last control parameters b_3^i; $i = 1,2,\ldots,m$, are obtained using after substituting b_0^i, b_1^i, b_2^i and eq. (23) in eq. (14) and take $x = 1$ with the aid of trapezoidal rule. When the values of the control parameters b_0^i, b_1^i, b_2^i and b_3^i are substituted into eq. (23), the solution to $O(x^3)$ is obtained.

Similarly, for the fourth order $B-spline$ functions and other higher orders.

From the above formulas we can concluded that as the order of $B-spline$ functions is increased, a large number of substitutions is required.

4. Convergence and Stability for the $B-spline$ functions

The $B-spline$ method is of general applicability, and it is the standard to which we compare the accuracy of the various other approximate methods for solving LVIODEs. It can be devised to have any specified degree of accuracy. We start by reformulating $B-spline$ functions in a form that is suitable for solving LVIODEs.

To find the error for $B-spline$ method, we can see for 1st order $B-spline$ functions that,

$$B^1(x) = b_0(1-x) + b_1 x, \qquad 0 \le x \le 1$$

$$\Rightarrow \quad B^1(x) = b_0 + (-b_0 + b_1)x$$

and by substituting $x = 0$, in this equation, we obtain $B^1(0) = b_0$. Then differentiate $B^1(x)$ with respect to x and substitute $x = 0$, yields

$$\frac{dB^1(x)}{dx}\Big|_{x=0} = -b_0 + b_1$$

Therefore,

$$B^1(x) = B^1(0) + \frac{dB^1(x)}{dx}\Big|_{x=0} x.$$

Similarly, for the 2nd order $B-spline$, we have

$$B^2(x) = b_0(1-x)^2 + 2b_1(1-x)x + b_2x^2,$$
$$0 \le x \le 1$$

$$= b_0 + (-2b_0 + 2b_1)x + (b_0 - 2b_1 + b_2)x^2$$

and $b_0 = B^2(0)$, $-2b_0 + 2b_1 = \frac{dB^2(x)}{dx}\big|_{x=0}$,

$$2!(b_0 - 2b_1 + b_2) = \frac{d^2B^2(x)}{dx^2}\Big|_{x=0}$$

That is,

$$B^2(x) = B^2(0) + \frac{dB^2(x)}{dx}\Big|_{x=0} x + \frac{1}{2!}\frac{d^2B^2(x)}{dx^2}\Big|_{x=0} x^2$$

For the 3rd order $B-spline$, we obtain

$$B^3(x) = B^3(0) + \frac{dB^2(x)}{dx}\Big|_{x=0} x + \frac{1}{2!}\frac{d^2B^2(x)}{dx^2}\Big|_{x=0}$$
$$x^2 + \frac{1}{3!}\frac{d^3B^2(x)}{dx^3}\Big|_{x=0} x^3$$

and soon, for the kth order $B-spline$, we have

$$B^k(x) = B^3(0) + \frac{dB^2(x)}{dx}\Big|_{x=0} x + \frac{1}{2!}\frac{d^2B^2(x)}{dx^2}\Big|_{x=0} 0$$
$$x^2 + \cdots \frac{1}{k!}\frac{d^kB^3(x)}{dx^k}\Big|_{x=0} x^k$$

or
$$B^k(x) = \sum_{j=0}^{k} \frac{d^jB^k(x)}{dx^j}\Big|_{x=0} \frac{x^j}{j!}$$

By convergence, we mean that the results of kth order $B-spline$ formula approaches to the exact solution as $k \to \infty$.

Assume $u \in c^k[0,1]$, that $u^{(k+1)}$ exists on $[0,1]$. for every $x \in [0,1]$ there exists a number $\varepsilon(x)$ between 0 and x with

$$u(x) = B^k(x) + R_k(x) \qquad (25)$$

where $B^k(x)$ is $B-spline$ functions that can be used to approximate $u(x)$, i.e.,

$$u(x) \approx B^k(x) = \sum_{j=0}^{k} \frac{d^j}{dx^j} B^k(0) \frac{x^j}{j!} \qquad (26)$$

The term $R_k(x)$ has the form

$$R_k(x) = \frac{d^{k+1}}{dx^{k+1}} B^k(\varepsilon(x)) \frac{x^{k+1}}{(k+1)!} \qquad (27)$$

Where $R_k(x)$ is called the reminder term (or truncation error) of order k associated with $B^k(x)$. It's the difference $u(x) - B^k(x)$ where $B^k(x)$ is used to approximate $u(x)$ near $x = 0$. When The infinite series obtained by taking the limit of $B^k(x)$ as $k \to \infty$ is called $B-spline$ series for $u(x)$ about 0.

Corollary:

If $B^k(x)$ is the $B-spline$ functions of order k given in eq. (26), then $B^k(0) = u(0)$ for all k

Since the accuracy of any given polynomial will generally decrease when we choose k large as the value of x moves away from the center 0; hence we must choose k large enough and restrict the maximum value $|x|$ so that the error does not exceed a specified bound.

If we choose the interval width to be $2R$ and 0 in the center (i.e., $|x| < R$), the absolute value of the error satisfies the relation

$$|error| = |R_k(x)| \le \frac{MR^{k+1}}{(k+1)!} \qquad (28)$$

where $M \le \max\left\{ \left| f^{(k+1)}(z) \right| : -R \le z \le R \right\}$. If N is fixed and the derivatives are uniformly bounded, the error bound in (28) is proportional to $R^{k+1}/(k+1)!$ and decreases if R goes to zero as N gets large.

We now discuss the stability of the $B-spline$ functions. We will show that the formula (16) is unconditionally stable.

- Using the first order $B-spline$, we set

$$b_0 = B^1(0)$$
$$b_1 = B^1(1) \qquad (29)$$

Eq. (29) can be rewritten as

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \end{pmatrix} = \begin{pmatrix} B^1(0) \\ B^1(1) \end{pmatrix}$$

where the vector $B = (B^1(0), B^1(1))^T$, is known vector.

- Using the second order $B-spline$, we get

$$b_0 = B^2(0)$$
$$b_1 = \frac{1}{2}\frac{dB^2(0)}{dx} + b_0 \qquad (30)$$
$$b_2 = B^2(1)$$

These equations can be put together into a single matrix equation as follows:

$$\begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} B^2(0) \\ \frac{1}{2}\frac{dB^2(0)}{dx} \\ B^2(1) \end{pmatrix}$$

where the vector

$$B = \left(B^2(0), \frac{1}{2}\frac{dB^2(0)}{dx}, B^2(1) \right)^T, \text{ is}$$

known vector.

- Successive applying kth order $B-spline$, eq. (16), will establish the following:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ a_{21} & 1 & 0 & 0 & \cdots & 0 \\ a_{31} & a_{32} & 1 & 0 & \cdots & 0 \\ \vdots & & & \ddots & & \vdots \\ a_{k-1,1} & \cdots & & a_{k-1,k-2} & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_{k-1} \\ b_k \end{pmatrix} = \begin{pmatrix} B^k(0) \\ c_1 \\ c_2 \\ \vdots \\ c_{k-1} \\ B^k(1) \end{pmatrix} \qquad (31)$$

or $Ab = c$

where the matrix A is $k+1\times k+1$ lower triangular matrix ; b and c are $k+1\times 1$ matrices.
Now the stability of (31) requires (Von Neumann condition) that all the eigen values of A lie on the interval $[-1,1]$. Since A is a lower triangular matrix, so we need only to show that $|a_{ii}|\leq 1$ for all $i=1,2,\ldots,I$, where a_{ii} denotes the diagonal element of A in row i. Since the diagonal elements of A is equal to 1, that is, $a_{ii}=1$ for all $i=1,2,\ldots,I$, therefore; we have shown that the scheme is unconditionally stable.

5. Numerical Example

The performance of the proposed method described in this paper will be compared using the following test xample with various orders of $B-spline$.
Consider the following LVIODEs

$$u_1'(x)+xu_1(x)=f_1(x)+\int_0^x \cos(x^2+t)u_1(t)dt+\int_0^x (x-t)u_2(t)dt$$
$$u_2'(x)-u_2(x)=f_2(x)+\int_0^x (1+x^2)u_2(t)dt \qquad (32)$$

where

$$f_1(x)=-\sin(x^2+x)-\frac{1}{2}x\cos x^2-\frac{1}{4}\sin(x^2+2x)+\frac{5}{4}\sin x^2+\sin x-x+x\cos x$$
$$f_2(x)=4\cos x+2x^2\cos x-2-2x^2-2\sin x$$

with $u_1(0)=2,\quad u_2(0)=0$
The exact solution is given by
$u_1(x)=1+\cos x,\quad u_2(x)=2\sin x$

When applying the $B-spline$ functions of orders $k=1,2,3,4$, the numerical solution to eq. (32) will be computed at the discretization points, x_i, where $x_i\in[0,1]$. In the present computations we will use $x_i=ih$; $i=0,1,\ldots,10$ with $h=0.1$.

The resulting variables $u_1(x_i)$ and $u_2(x_i)$; $i=1,2,\ldots,10$ are displayed in tables (1) and (2) respectively. Also, the exact solution is listed in the tables for both $u_1(x)$ and $u_2(x)$.

What is important in practice is the speed of convergence associated with various k (the order of $B-spline$ functions). In general, we want k be as large as possible. Here , the convergence is calculated depending on the least square errors (L.S.E.).
Let the least square errors for u_1 denoted by $(\text{L.S.E.})_{u_1}$ and the least square errors for u_2 denoted by $(\text{L.S.E.})_{u_2}$, then the least square errors for eq. (32) is equal to

$$\text{L.S.E.}=\max\left((\text{L.S.E.})_{u_1},(\text{L.S.E.})_{u_2}\right)$$

The results in table (3) show that, the error decreases substantially as the spline order is increased.

Table (1):The resulting variable $u_1(x)$

x	Exact $u(x) = 1 + \cos x$	Spline Orders 1	2	3	4
0	2.000000000	2.000000000	2.000000000	2.000000000	2.000000000
0.1	1.995004165	1.954014275	1.995401427	1.995040142	1.995004014
0.2	1.980066577	1.908028550	1.981605710	1.980331142	1.980064228
0.3	1.955336489	1.862042825	1.958612847	1.956083854	1.955325156
0.4	1.921060994	1.816057100	1.926422840	1.922569136	1.921027654
0.5	1.877582561	1.770071375	1.885035687	1.880017843	1.877508921
0.6	1.825335614	1.724085650	1.834451390	1.828670834	1.825202500
0.7	1.764842187	1.678099926	1.774669948	1.768768963	1.764638274
0.8	1.696706709	1.632114201	1.705691361	1.700553088	1.696442471
0.9	1.621609968	1.586128476	1.627515628	1.624264065	1.621337659
1	1.540302305	1.540142751	1.540142751	1.540142751	1.540142751

Table (2)):The resulting variable $u_2(x)$

x	Exact $u_2(x) = 2\sin x$	Spline Orders 1	2	3	4
0	0	0	0	0	0
0.1	0.199666833	0.168251387	0.196825138	0.199682513	0.199668251
0.2	0.397338661	0.336502774	0.387300554	0.397460110	0.397358688
0.3	0.591040413	0.504754161	0.571426248	0.591427874	0.591128362
0.4	0.778836684	0.673005548	0.749202219	0.779680887	0.779072355
0.5	0.958851077	0.841256935	0.920628467	0.960314233	0.959323783
0.6	1.129284946	1.009508322	1.085704993	1.131422996	1.130053797
0.7	1.288435374	1.177759709	1.244431796	1.291102257	1.289471580
0.8	1.434712181	1.346011096	1.396808877	1.437447101	1.435824348
0.9	1.566653819	1.514262483	1.542836235	1.568552611	1.567397350
1	1.682941969	1.682513870	1.682513870	1.682513870	1.682513870

Table (3):Error test

Spline Orders	1	2	3	4
$(L.S.E.)_{u_1}$	6.1364732×10^{-2}	3.92842×10^{-4}	5.723733×10^{-5}	2.353996×10^{-7}
$(L.S.E.)_{u_2}$	7.4370141×10^{-2}	8.672188×10^{-3}	2.597061×10^{-5}	3.925006×10^{-6}
(L.S.E.)	7.4370141×10^{-2}	8.672188×10^{-3}	5.723733×10^{-5}	3.925006×10^{-6}

Notice that, every order k increases the accuracy by one digit.

6. Conclusion

We have presented an approximate method to solve LVIODEs. The solution is obtained by introducing $B-spline$ functions which we showed to be unconditionally stable and has the property that the final global error is of order $o(x^{k+1})$, where k is the order of $B-spline$ functions. Hence k can be chosen as large as necessary to make this error as small as desired. That is the accuracy of $B-spline$ functions is increased when we choose k large.

Numerical results were presented to verify our theoretical results and to demonstrate the usefulness of the method.

References

[1] Lashmikantham, V. and Rama, M. M. R.,1995, Theory of Integro-Differential Equations, Gordon and Breach Science Publisher.

[2] Lutterkort, D., 1999, Computing Linear Envelopes for Uniform $B-spline$ Curves, Internet.

[3] $B-spline$. Nov. 16, 2001,Curves, Internet, Lesson (15).

[4] $B-spline$ Curves, Internet, www.math.hmc.edu/~/math.142/mellon/Application-to-GAGD/ $B-spline$ Curves. Html\-9k, 2002.

[5] Cox, M.G. .1972, The Numerical Evaluation of $B-splines$, J. Inst. Maths Applics 10, 134-149.

[6] Burden, R.L. and Faires, J.D., 1997, Numerical Analysis, An Introduction, Thomson Publishing Company (ITP.

[7] Mathews, J.H. and Fink, K.D., 1999, Numerical Methods Using MATLAB, Prenice-Hall, Inc.

حل نوع من منظومة معادلات فولتيرا التفاضلية-التكاملية باستخدام دوال الثلمة

عمر محمد الفاعور*

* أ.م.د./ الجامعة التكنولوجية/ قسم العلوم التطبيقية

المستخلص:

في هذا البحث، أقترحنا تقريب لمنظومة معادلات فولتيرا التفاضلية-التكاملية الخطية من النوع الأول. فـي هـذا التقريـب، قـد تـم أولا تشـكيل دوال الثلمـة باسـتخدام المصـفوفة، ثـم ناقشـنا الأقتـراب والأسـتقرارية لدوال الثلمة.

نتائج عددية و المتضمنة حل منظومة معادلات فولتيرا التفاضلية-التكاملية الخطية من النوع الأول ترينا كفاءة الطريقة المقترحة باستخدام دوال الثلمة .