Eng. & Tech. Journal, Vol.29, No.9, 2011

Expansion Methods for Solving Linear Integral Equations with Multiple Time Lags Using B-Spline and Orthogonal Functions

Atheer Jawad Kadhim

Received on: 3 /5/ 2010
Accepted on: 7/4/ 2011

Abstract

The main portion of the paper is devoted for constructing approximated solutions of linear integral equations with multiple time lags (IEMTL) using two different types of basis functions B-spline functions and Orthogonal functions containing (Laguerre and Hermite) with the aid of expansion methods (collocation method, Partition method and Galerkin's method). Algorithms with the aid of Matlab language are derived for treating these equations using expansion methods. Comparison between the exact and approximated results of these methods with the aid of basis functions are given via two test examples and accurate results are achieved.

Keywords: Integral equations with multiple time lags, Expansion methods, B-spline functions and Orthogonal functions.

طرائق التوسيع لحل المعادلات التكاملية الخطية ذات زمن متعدد التباطؤ باستخدام الدوال الثلمة التوصيلية و الدوال المتعامدة

الخلاصة

الفكرة الأساسية للبحث مكرسة لبناء الحلول التقريبية للمعادلات التكاملية الخطية ذات زمن متعدد التباطؤ مستخدماً نوعين مختلفين من الدوال الاساسية: الدوال الثلمة التوصيلية و الدوال المتعامدة و المتضمنة (لاكيرو هيرمايت) معتمدا على طرائق التوسيع والمتضمنة (طريقة التجميع ،طريقة التجزئة و طريقة كالكن). حيث تم اشتقاق خوارزميات تمت برمجتها بلغة (Matlab) لمعالجة هذه المعادلات التكاملية باستخدام طرائق التوسيع اعلاه. كما تمت مقارنة النتائج الحقيقية و التقريبية لهذه الطرائق التوسيعية مع الدوال الأساسية لهذه المعادلات من خلال مثالين وقد تم الحصول على نتائج دقيقة.

1. Introduction

One of the most important and applicable subjects of applied mathematics, and in developing modern mathematics is the integral equations. The names of many modern mathematicians notably, Volterra, Fredholm, Cauchy and others are associated with this topic [1]. The name integral equation was introduced by Bois-Reymond in 1888 [2].

The most resent kind of equation that worth studying is the delay integral equation. These equations have many applications like: a model to explain the observed periodic out breaks of certain infection diseases [3]. Another application is the electromagnetic inverse scattering problem in a medium with

* Applied Science Department, University of Technology/ Baghdad

https://doi.org/10.30684/etj.29.9.3
2412-0758/University of Technology-Iraq, Baghdad, Iraq

discontinuous change in conductivity [4].

To facilitate the presentation of the material that followed, a brief review of some background on the linear integral equations with multiple time lags and their types are given in the following section.

2. Linear Integral Equations with Multiple Time Lags (L-IEMTL) [5, 6, 7]

The significance of these equations lies in their ability to describe processes with retarded (delay) time which may appear in the unknown function u (t) involved in the integrand or may appear in the unknown function $u(t)$ in the left hand side of the equation or may appear in one of the limits of the integrations.

The integral equations with multiple time lags (IEMTL) have two lags τ_1 and τ_2 such that $\tau_1, \tau_2 \in R,\ \tau_1$ *and* $\tau_2 > 0$ and they

Can be classified into the following cases:

1. If τ_1 appears in the unknown function $u(t)$ outside the integral sign and τ_2 appears in the unknown function $u(t)$ inside the integral sign such that:

$$h(t)u(t-\tau_1) = g(t) + \int_a^{b(t)} k(t,x)u(x-\tau_2)dx \cdots (1)$$

If τ_1 appears in the unknown function $u(t)$ outside the integral sign and τ_2 appears in one of the limits of integration such that:

$$h(t)u(t-\tau_1) = g(t) + \int_a^{\tau_2} k(t,x)u(x)dx \cdot \quad ....(2)$$

Or

$$h(t)u(t-\tau_1) = g(t) + \int_{\tau_2}^{b(t)} k(t,x)u(x)dx \cdots (3)$$

If τ_1 appears in the unknown function $u(t)$ inside the integral sign and τ_2 appears in one of the limits of integration:

$$h(t)u(t) = g(t) + \int_a^{\tau_2} k(t,x)u(x-\tau_1)dx \ ..(4)$$

Or

$$h(t)u(t) = g(t) + \int_{\tau_2}^{b(t)} k(t,x)u(x-\tau_1)dx \quad (5)$$

where $h(t)$ and $g(t)$ are known functions of t , $k(t,x)$ is called the kernel of the IEMTL.

Remarks [6,8]

- If $h(t) = 0$ then the above equations called IEMTL of the first kind.
- If $h(t) = 1$ then the above equations called IEMTL of the second kind.
- If $g(t) = 0$ then the above equations called homogeneous IEMTL otherwise if $g(t) \neq 0$ then they called non-homogeneous IEMTL.
- If $b(t) = t$ then the above equations called Volterra integral equations with multiple time lags while if $b(t) = b,\ b$ is a constant then the above equations called Fredholm integral equations with multiple time lags

3. Expansion Methods

Expansion methods or weighted residual methods [9,10] are presented by considering the following functional equation:

$$L[u(t)] = g(t) \quad t \in D \quad \dots (6)$$

Where L denotes an operator which maps a set of functions, say U, into a set of function, say G, such that u∈ U , g∈ G and D is a prescribed domain.

The epitome of the expansion method is to approximate the unknown solution $u(t)$ of eq.(6) by a set of known functions as:

$$u(t) \cong u_N(t) = \sum_{i=0}^{N} c_i \psi_i(t) \quad \dots (7)$$

where $N > 0$ and $c_0, c_1, \dots, c_N$ are (N+1) unknown coefficients. The functions $\psi_i(t)$ are chosen in this work to be B-spline functions or orthogonal functions, which are prescribed in section (4).

An approximated solution $u_N(t)$ given by eq.(7) will not, in general, satisfy eq.(6) exactly. Therefore a term, say $E_N(t)$, called the residual is associated with such an approximated solution and is defined by:

$$E_N(t) = L[u_N(t)] - g(t) \quad \dots (8)$$

The residual $E_N(t)$ depends on t as well as on the way that the parameters c_i's be chosen.

It is obvious that when $E_N(t) = 0$, then the exact solution is obtained which is difficult to be achieved, therefore we shall try to minimize $E_N(t)$ in some sense.

In the expansion method the unknown parameters c_i's are chosen to minimize the residual $E_N(t)$ by setting its weighted integral equal to zero, i.e.

$$\int_D w_j E_N(t)dt = 0 \quad j = 0,1,\dots,N \quad (9)$$

Where w_j is a prescribed weighting functions, $t \in D$ and D is a prescribed domain. The technique based on eq.(9) is called weighted residual method. Different choices of w_j yield different approximate solutions. The expansion methods that will be discussed in this work are: collocation, partition and Galerkin methods.

4. Basis Functions

In this work, the choice of basis functions $\psi_i(t)$ are:

4.1 B-Spline Functions

The n^{th} order B-splines as appropriately scaled n^{th} divided difference of truncated power function; these functions have several mathematical definitions [11].

B-splines were introduced around 1940's in the context of approximation theory [11]. Schoenberg [12] introduced the B-spline in 1949 and B-splines have been applied to geometric modeling since 1970's [11]. According to Schoenberg approach, B-spline means spline basis and the letter B in B-spline stands for basis [11].

The B-spline can be defined as [12, 13]:

$$B_{k,n}(t) = \binom{n}{k} t^k (1-t)^{n-k} \quad k \ge 0, n \ge 0,$$

$$t \in (-\infty, \infty) \quad \dots (10)$$

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 $B_{k,n}(t) = 0$ if $k < 0$ or $k > n$.

4.1.1 Some Types of B-Spline Functions [11,12]

1. The Constant B-spline $B_{k,0}(t)$:

The constant B-spline or B-spline of order 0 is the simplest spline. It is defined on one knot span only and is not even continued on the knots.

$$B_{k,o}(t) = \begin{Bmatrix} 1 & if\ t_k \le t < t_{k+1} \\ 0 & otherwise \end{Bmatrix}$$

2. The Linear B-spline $B_{k,1}(t)$:

The linear B-spline or the first order B-spline is defined on two consecutive knot spans and is continued on the knots.

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

3. Quadratic B-spline $B_{k,2}(t)$:

Quadratic B-spline (or the 2nd order B-spline) with uniform knot-vector is a commonly used form of B-spline, where

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

4. Cubic B-spline $B_{k,3}(t)$

Cubic B-spline (or the 3rd order B-spline) with uniform knot-vector is the most commonly used form of B-spline, where

$$B_{0,3}(t) = (1-t)^3 \quad , \quad B_{1,3}(t) = 3t(1-t)^2$$
$$, B_{2,3}(t) = 3t^2(1-t) \quad and \quad B_{3,3}(t) = t^3$$

4.2 Orthogonal Functions [14,15]

Orthogonal functions (or polynomials) are of great importance in many area of mathematics particularly approximation theory. Numerous articles and books have been written about this topic. There are several kinds of orthogonal polynomials. In particular we shall introduced (Laguerre and Hermite) polynomials.

4.2.1 Laguerre Polynomials

The Laguerre polynomials $L_n(x)$ are an important set of orthogonal functions over the interval [0, ∞) .

The general form of these polynomials is:

$$L_{n+1}(x) = \left(\frac{2n+1-x}{(n+1)}\right)L_n(x) - \left(\frac{n}{(n+1)}\right)L_{n-1}(x) \quad n \ge 1 \qquad \ldots (11)$$

where $L_0(x) = 1$ and $L_1(x) = 1 - x$.

4.2.2 Hermite Polynomials

The Hermite polynomials $H_n(x)$ are an important set of orthogonal functions over the interval (-∞, ∞) and the general form of these polynomials is:

$$H_{n+1}(x) = -2xH_n(x) - 2nH_{n-1}(x) \quad n \ge 1 \qquad \cdots (12)$$

where $H_0(x) = 1$ and $H_1(x) = -2x$.

5. The Solution of Linear IEMTL Using Expansion Methods with The Aid of B-Spline and Orthogonal Functions

Expansion methods are one of the most efficient methods used to solve integral equations without time lag. In this section, expansion methods with the aid of B-Spline functions and orthogonal functions are candidates to find the approximated solutions for IEMTL as follows:

Consider the linear IEMTL of the second kind:

$$u(t-\tau_1) = g(t) + \int_a^{b(t)} k(t,x)u(x-\tau_2)dx \quad t \in [a, b(t)] \qquad \ldots (13)$$

where $\tau_1, \tau_2 \in R$, τ_1 and $\tau_2 > 0$.

Expansion methods are based on approximating the unknown function $u(t)$ by eq.(7) which is:

$$u(t) \cong u_N(t) = \sum_{i=0}^{N} c_i \psi_i(t)$$

where $\psi_i(t)$ are chosen to be one of the following:

(a) B-spline functions: $\psi_i(t) = B_{i,N}(t)$

(b) Orthogonal functions which are:

- Laguerre polynomials: $\psi_i(t) = L_i(t)$
- Hermite polynomials: $\psi_i(t) = H_i(t)$

By substituting eq.(7) into eq.(13), one gets the following formula:

$$\sum_{i=0}^{N} c_i \psi_i(t-\tau_1) = g(t) + \int_a^{b(t)} k(t,x)\left[\sum_{i=0}^{N} c_i \psi_i(x-\tau_2)\right] dx \quad (14)$$

Then,

$$\sum_{i=0}^{N} c_i \left(\psi_i(t-\tau_1) - \int_a^{b(t)} k(t,x)\psi_i(x-\tau_2)\,dx \right) = g(t) \quad (15)$$

Hence, the residual equation $E_N(t)$ in eq.(9) for eq.(13) is defined by:

$$E_N(t) = \sum_{i=0}^{N} c_i \left(\psi_i(t-\tau_1) - \int_a^{b(t)} k(t,x)\psi_i(x-\tau_2)\,dx \right) - g(t)$$

(16)

5.1 The Solution of Linear IEMTL Using Collocation Method

In Collocation method [16] the weighting functions are chosen to be:

$$w_j = \begin{cases} 1 & if\ t = t_j \\ 0 & otherwise \end{cases} \quad \text{... (17)}$$

where the fixed points $t_j \in D \ \ and \ \ j = 0,1,...,N$ are called collocation points.

Inserting eq.(17) in eq.(9) gives:

$$\int_D w_j E_N(t)\,dt = E_N(t_j)\int_D w_j\,dt = 0 \rightarrow \quad (18)$$
$$E_N(t_j) = 0 \quad j = 0,1,...,N$$

This leads eq.(16) to:

$$E_N(t_j) = \sum_{i=0}^{N} c_i \left(\psi_i(t_j - \tau_1) - \int_a^{b(t_j)} k(t_j,x)\psi_i(x-\tau_2)\,dx \right) - g(t_j) = 0$$

Hence,

$$\sum_{i=0}^{N} c_i \left(\psi_i(t_j - \tau_1) - \int_a^{b(t_j)} k(t_j,x)\psi_i(x-\tau_2)\,dx \right) = g(t_j)$$

$j=0,1,\ldots,N$ … (19)

So, by expanding and simplifying eq.(19), we have (N+1) simultaneous equations with (N+1) unknown coefficients $c_0, c_1, \ldots, c_N$.

Hence, eq.(19) can be written in matrix form as $DC=G$, where

$$\left. \begin{array}{l} D = \begin{bmatrix} d_{00} & d_{01} & \cdots & d_{0N} \\ d_{10} & d_{11} & \cdots & d_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ d_{N0} & d_{N1} & \cdots & d_{NN} \end{bmatrix}, \\ C = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_N \end{bmatrix} \quad and \quad G = \begin{bmatrix} g(t_0) \\ g(t_1) \\ \vdots \\ g(t_N) \end{bmatrix} \end{array} \right\} \quad (20)$$

where

$$d_{j,i} = \left(\psi_i(t_j - \tau_1) - \int_a^{b(t_j)} k(t_j,x)\psi_i(x-\tau_2)\,dx \right)$$

for $j = 0,1,\ldots,N$ and $i = 0,1,\ldots,N$.

Then, using Gauss elimination method to find the coefficients c_i's $, i = 0,1,...,N$ which satisfy eq.(7) (the approximate solution $u_N(t)$ of eq.(13)).

The solution of linear **IEMTL** using ***collocation method*** with the aid of B-Spline functions and orthogonal functions can be summarized by the following algorithm:

IEMTL-CM Algorithm

INPUT

- N: (the number of $\psi_i(t), i = 0,1,...,N$)
- $t_0, t_1, ..., t_N$: (the collocation points).
- a & $b(t)$: (the limits of the integral of IEMTL) .
- The function $g(t)$ of IEMTL.
- The kernel $k(t,x)$ of the IEMTL.

OUTPUT

- c_i's $, i = 0,1,...,N$:(the unknown coefficients of eq.(7)).
- $u_N(t)$: (the approximate solution of IEMTL)

Step 1: Select the basis function $\psi_i(t)$ for i = 0, 1, ..., N.

Step 2: Set $u_N(t) = c_0\psi_0(t) + c_1\psi_1(t) + \cdots + c_N\psi_N(t)$

Step 3: Set $j = 0$

Step 4: Compute eq.(19)

Step 5: Put $j = j+1$

Step 6: If $j = N+1$ then stop and go to (step 7).

Else go to (step 4)

Step 7: Express the (N+1) simultaneous equations in step(4) by matrix form $DC=G$ as in eq.(20).

Step 8: Use Gauss elimination method for finding the coefficients c_i's $, i = 0,1,...,N$ which give the solution $u_N(t)$ in (step 2).

5.2 The Solution of Linear IEMTL Using Partition Method

In partition method [17] the domain D is divided into N non-over lapping sub domains D_j , j=0,1,…,N , N >0 and the weighting functions w_j in eq.(9) are defined as :

$$w_j = \begin{cases} 1 & if\ t \in D_j \\ 0 & if\ t \notin D_j \end{cases} \qquad \ldots (21)$$

Then, eq.(13) is satisfied on the average in each of N sub domains D_j .

By substituting eq.(16) and eq.(21) into eq.(9) yields:

$$\int_{D_j} \left[\sum_{i=0}^{N} c_i \left(\psi_i(t-\tau_1) - \int_a^{b(t)} k(t,x)\psi_i(x-\tau_2)\,dx \right) - g(t) \right] dt = 0$$
$$D_j \in D \quad and \quad j = 0,1,...,N \qquad \ldots (22)$$

Hence,

$$\sum_{i=0}^{N} c_i \int_{D_j} \left(\psi_i(t-\tau_1) - \int_a^{b(t)} k(t,x)\psi_i(x-\tau_2)\,dx \right) dt = \int_{D_j} g(t)dt$$
$$D_j \in D \quad and \quad j = 0,1,...,N \qquad \ldots\ldots(23)$$

So, by expanding and simplifying eq.(23), we have (N+1) simultaneous equations with (N+1) unknown coefficients $c_0, c_1, \ldots, c_N$.

Hence, eq.(23) can be written in matrix form as $DC=G$, where

$$\left. \begin{array}{l} D = \begin{bmatrix} d_{00} & d_{01} & \cdots & d_{0N} \\ d_{10} & d_{11} & \cdots & d_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ d_{N0} & d_{N1} & \cdots & d_{NN} \end{bmatrix}, \quad C = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_N \end{bmatrix} \\ and \quad G = \begin{bmatrix} \int_{D_0} g(t)dt \\ \int_{D_1} g(t)dt \\ \vdots \\ \int_{D_N} g(t)dt \end{bmatrix} \end{array} \right\} \qquad \ldots (24)$$

and

$$d_{j,i} = \int_{D_j} \left(\psi_i(t-\tau_1) - \int_a^{b(t)} k(t,x)\psi_i(x-\tau_2)\,dx \right) dt$$

for $j = 0,1,\ldots,N$ and $i = 0,1,\ldots,N$.

Then, using Gauss elimination method to find the coefficients c_i's $, i = 0,1,...,N$ which give eq.(7) (the approximate solution $u_N(t)$ of eq.(13)).

The solution of linear **IEMTL** using ***partition method*** with the aid of B-Spline functions and orthogonal functions can be summarized by the following algorithm:

Iemtl-Pm Algorithm

Input

- N: (the number of $\psi_i(t), i = 0,1,...,N$)
- a & $b(t)$: (the limits of the integral of IEMTL) .
- The function $g(t)$ of IEMTL.
- The kernel $k(t,x)$ of the IEMTL.

OUTPUT

- c_i's $, i = 0,1,...,N$:(the unknown coefficients of eq.(7)).
- $u_N(t)$: (the approximate solution of IEMTL)

__Step 1:__ Select the basis function $\psi_i(t)$ for $i = 0, 1, ..., N$.

__Step 2:__ Set $u_N(t) = c_0\psi_0(t) + c_1\psi_1(t) + \cdots + c_N\psi_N(t)$

__Step 3:__ Set $j = 0$

__Step 4:__ Compute eq.(23)

__Step 5:__ Put $j = j+1$

__Step 6:__ If $j = N+1$ then stop and go to (step 7).

Else go to (step 4)

__Step 7:__ Express the $(N+1)$ simultaneous equations in step (4) by matrices form $DC=G$ as eq.(24).

__Step 8:__ Use Gauss elimination method for finding the coefficients c_i's $, i = 0,1,...,N$ which satisfy the solution $u_N(t)$ in (step 2).

5.3 The Solution of Linear IEMTL Using Galerkin's Method

In Galerkin's method [18] the weight functions w_j in eq.(9) are defined as :

$$w_j(t) = \frac{\partial u_N(t)}{\partial c_j} \qquad j = 0,1,\ldots,N$$

Since $u_N(t) = \sum_{i=0}^{N} c_i\psi_i(t)$

Then,

$$w_j(t) = \frac{\partial \sum_{i=0}^{N} c_i\psi_i(t)}{\partial c_j} = \psi_j(t) \qquad j = 0,1,\ldots,N \qquad \ldots (25)$$

Substituting eq.(16) and eq.(25) into eq.(9) yields:

$$\int_D \psi_j(t)\left[\sum_{k=0}^{N} c_k\left(\psi_k(t-\tau_1) - \int_a^{b(t)} k(t,x)\psi_k(x-\tau_2)\,dx\right) - g(t)\right]dt = 0$$

$$j = 0,1,...,N \qquad \ldots (26)$$

Hence,

$$\sum_{i=0}^{N} c_i \int_D \psi_j(t)\left(\psi_i(t-\tau_1) - \int_a^{b(t)} k(t,x)\psi_i(x-\tau_2)\,dx\right)dt = \int_D \psi_j(t)g(t)dt \qquad j = 0,1,...,N \qquad \ldots (27)$$

So, by expanding and simplifying eq.(27), we have $(N+1)$ simultaneous equations with $(N+1)$ unknown coefficients $c_0, c_1, \ldots, c_N$.

Hence, eq.(27) can be written in matrix form as $DC=G$, where

$$D=\begin{bmatrix} d_{00} & d_{01} & \dots & d_{0N} \\ d_{10} & d_{11} & \dots & d_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ d_{N0} & d_{N1} & \dots & d_{NN} \end{bmatrix},\ C=\begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_N \end{bmatrix}$$

$$and \quad G=\begin{bmatrix} \int_D \psi_0(t)g(t)dt \\ \int_D \psi_1(t)g(t)dt \\ \vdots \\ \int_D \psi_N(t)g(t)dt \end{bmatrix} \quad \dots (28)$$

where

$$d_{j,i}= \int_D \psi_j(t)\left(\psi_i(t-\tau_1)-\int_a^{b(t)} k(t,x)\psi_i(x-\tau_2)\,dx\right)dt$$

for $j=0,1,\dots,N$ and $i=0,1,\dots,N$.

Then, using Gauss elimination method to find the coefficients c_i's $,i=0,1,...,N$ which give eq.(7) (the approximate solution $u_N(t)$ of eq.(13)).

The solution of linear **IEMTL** using ***G****alerkin's* ***m****ethod* with the aid of B-Spline functions and orthogonal functions can be summarized by the following algorithm:

Iemtl-Gm Algorithm

Input

- N: (the number of $\psi_i(t), i=0,1,...,N$)
- a & $b(t)$: (the limits of the integral of IEMTL) .
- The function $g(t)$ of IEMTL.
- The kernel $k(t,x)$ of the IEMTL.

OUTPUT

- c_i's $,i=0,1,...,N$:(the unknown coefficients of eq.(7)).
- $u_N(t)$: (the approximate solution of IEMTL)

<u>Step 1:</u> Select the basis function $\psi_i(t)$ for i = 0, 1, ..., N.

<u>Step 2:</u> Set $u_N(t)=c_0\psi_0(t)+c_1\psi_1(t)+\dots+c_N\psi_N(t)$

<u>Step 3:</u> Set $j=0$

<u>Step 4:</u> Compute eq.(27)

<u>Step 5:</u> Put $j=j+1$

<u>Step 6:</u> If $j=N+1$ then stop and go to (step 7).
Else go to (step 4)

<u>Step 7:</u> Express the (N+1) simultaneous equations in step(4) by matrix form $DC=G$ as in eq.(28).

<u>Step 8:</u> Use Gauss elimination method for finding the coefficients c_i's $,i=0,1,...,N$ which give the solution $u_N(t)$ in (step 2).

6. Test Examples :

<u>Example (1):</u>

Consider the following Volterra integral equation with multiple time lags:

$$u(t-\tfrac{1}{2})=\left(\frac{1}{2}+t-\frac{t^4}{3}\right)+\int_0^t t\,x\,u(x-1)\,dx \qquad t\in[0,2] \qquad \dots (29)$$

The exact solution of eq.(29) is: $u(t)=t+1 \quad 0\le t\le 2$.

Assume that the approximate solution of eq.(29) is:

1. $u(t)\cong u_1(t)=\sum_{i=0}^{1} c_i B_{i,1}(t)$
2. $u(t)\cong u_1(t)=\sum_{i=0}^{1} c_i L_i(t)$
3. $u(t)\cong u_1(t)=\sum_{i=0}^{1} c_i H_i(t)$

When the algorithms (IEMTL-CM, IEMTL-PM and IEMTL-GM) are applied, table (1) presents the comparison between the exact and collocation, partition and Galerkin methods with the aid of basis functions containing (B-spline, Laguerre and Hermite) functions, for eq.(29) depending on least square error (L.S.E.) where m=10, h=0.2, $t_j = jh$, $j = 0,1,...,m$.

Example (2)

Consider the following Fredholm integral equation with multiple time lags:

$$u(t-0.2) = \left(e^{(t-\frac{1}{5})} - te + t - 1\right) + \int_0^{\tau_2} (t+x)\,u(x)\,dx$$

$t \in [0,1]$ … (30)

where $\tau_2 = 1$ and the exact solution of eq.(30) is:

$$u(t) = e^t \qquad 0 \le t \le 1 .$$

Assume that the approximate solution of eq.(30) is:

1. $u(t) \cong u_5(t) = \sum_{i=0}^{5} c_i B_{i,5}(t)$
2. $u(t) \cong u_5(t) = \sum_{i=0}^{5} c_i L_i(t)$
3. $u(t) \cong u_5(t) = \sum_{i=0}^{5} c_i H_i(t)$

When algorithms (IEMTL-CM, IEMTL-PM and IEMTL-GM) are applied, table (2) presents the comparison between the exact and collocation, partition and Galerkin methods with the aid of basis functions containing (B-spline, Laguerre and Hermite) functions, for eq.(30) depending on least square error (L.S.E.) where m=10, h=0.1, $t_j = jh$, $j = 0,1,...,m$.

Conclusions

The approximated solutions using three types of expansion methods containing (collocation, partition and Galerkin) with the aid of two different types of basis functions (B-spline functions and orthogonal functions) have been obtained for two examples. The results showed a marked improvement in the least square errors and the following conclusion points are listed:

1. In terms of the results, Galerkin's method gave more accurate results than collocation and partition methods, see table (2).
2. For basis functions, B-spline functions gave more accurate results than orthogonal functions (Laguerre and Hermite polynomials), see table (2).
3. For orthogonal functions, Laguerre polynomials gave more accurate results than Hermite polynomials.
4. Good approximation of the solution depends on :
   - The number N of basis functions where as N increases, the error term approaches to zero.
   - The existence of the roots x^j, *j=0,1,…,N* in orthogonal polynomials, (i.e. as it increases as in Laguerre polynomials the L.S.E. approaches to zero).

References

[1] Jerri, A.J., Introduction to Integral Equations with Applications; 1st edition, Marcel Dekker; USA, 1985.

[2] Burton, T.A., Integral Equation with Delay; Aeta Math hung 72, No 3, PP.233-242, 1998.

[3] Smith, H., On Periodic Solution of Delay Integral Equations Modeling Epidemics, Jr. Math Biology, No.4, pp.69-80, 1997.

[4] Kamen, D.E., Khargonekar, P.P. and Tannenbaum, A., Proper Factorizations and Feedback Control of Linear Time-Delay Systems. International Jr. of Control, No.43, PP.837–857, 2006.

[5] Akram H., About the Singular Operator In Integral Equations, M.Sc. Thesis, Collage of science, Al-Nahrain University, 1999.

[6] Burton T., Integral Equations with Delay, Acta Math, Hung, Vol.27, No.3, PP.233-242, 1996.

[7] Ali, H.A., Approximated Solutions of Linear Delay Integro-Differential Equations, M.Sc. Thesis, Applied science department, University of Technology, IRAQ, 2006.

[8] Abdul Hameed, F.T., Numerical Solutions of Fredholm Integro-Differential Equations Using Spline Functions, M.Sc. Thesis, Applied science department, University of Technology, IRAQ, 2002.

[9] AL-Marie, N.K.; Variation Formulation of Delay Differential Equations, M.Sc.Thesis college of Education, Ibn AL-Haitham, Baghdad University, 2001.

[10] Davis A., Expansion methods for Solving ODE and PDE, http://www.cs.4t.ee/Ntoo mas/linalg/ dv2/nod28.htm/2004.

[11] Matrin H., B-Splines Basis Functions, www.cs.mtu.edu/~chene/courses/cs 3621/notes/spline/B-spline/basis.htm1-18k. 2003.

[12] Hassen, S.S., Numerical Algorithms for Solving Optimal Control Problems, M.Sc. Thesis, Applied science department, University of Technology, 2006.

[13] Hsu, Y.L., B-Spline Curves, http://designer.mech.yzu.edu.tw/. 2002.

[14] Byung-Kookim; Piecewise linear Dynamic systems with time Delay, Ph.D. Thesis, California Institute of Technology, Pasadena, California, 1972.

[15] San, G., Orthogonal Functions, http://mathword/orthogonal polynomials/htm., 2002.

[16] Engelborghs, K., Luzyanina, T., Roose, D., Collocation Methods for the Computation of Periodic Solutions of Delay Differential Equations, SIAM Jr. Sci.Comput. Vol.22, No.5, PP.1593-1609, 2000.

[17] Kareem, R.S., Approximated Treatment of Higher-Order Linear Volterra Integro Differential Equations, M.Sc. Thesis, Applied science department, University of Technology, 2003.

[18] Gelfand M.&Famines S., Methods of Numerical Approximation, Oxford University Computing Laboratory, 2006.

Table (1) The Solution of eq.(29) for example(1)

t	*Exact*	*Collocation (IEMTL-CM) Algorithm*			*Partition (IEMTL-PM) Algorithm*			*Galerkin (IEMTL-GM) Algorithm*		
		B-spline	*Laguerre*	*Hermite*	*B-spline*	*Laguerre*	*Hermite*	*B-spline*	*Laguerre*	*Hermite*
0	1	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
0.2	1.2	1.2000	1.2000	1.2000	1.2000	1.2000	1.2000	1.2000	1.2000	1.2000
0.4	1.4	1.4000	1.4000	1.4000	1.4000	1.4000	1.4000	1.4000	1.4000	1.4000
0.6	1.6	1.6000	1.6000	1.6000	1.6000	1.6000	1.6000	1.6000	1.6000	1.6000
0.8	1.8	1.8000	1.8000	1.8000	1.8000	1.8000	1.8000	1.8000	1.8000	1.8000
1	2	2.0000	2.0000	2.0000	2.0000	2.0000	2.0000	2.0000	2.0000	2.0000
1.2	2.2	2.2000	2.2000	2.2000	2.2000	2.2000	2.2000	2.2000	2.2000	2.2000
1.4	2.4	2.4000	2.4000	2.4000	2.4000	2.4000	2.4000	2.4000	2.4000	2.4000
1.6	2.6	2.6000	2.6000	2.6000	2.6000	2.6000	2.6000	2.6000	2.6000	2.6000
1.8	2.8	2.8000	2.8000	2.8000	2.8000	2.8000	2.8000	2.8000	2.8000	2.8000
2	3	3.0000	3.0000	3.0000	3.0000	3.0000	3.0000	3.0000	3.0000	3.0000
L.S.E.		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table (2) The Solution of eq.(30) of example(2)

t	*Exact*	*Collocation (IEMTL-CM) Algorithm*			*Partition (IEMTL-PM) Algorithm*			*Galerkin (IEMTL-GM) Algorithm*		
		B-spline	*Lag.*	*Herm.*	*B-spline*	*Lag.*	*Hermite*	*B-spline*	*Lag.*	*Herm.*
0	1.0000	1.0000	1.0000	1.0000	1.0000	0.9999	1.0003	1.0000	1.0000	1.0001
0.1	1.1052	1.1052	1.1052	1.1058	1.1052	1.1051	1.1060	1.1052	1.1052	1.1058
0.2	1.2214	1.2214	1.2215	1.2225	1.2214	1.2214	1.2226	1.2214	1.2214	1.2224
0.3	1.3499	1.3499	1.3499	1.3513	1.3499	1.3499	1.3514	1.3499	1.3499	1.3512
0.4	1.4918	1.4918	1.4919	1.4935	1.4918	1.4919	1.4936	1.4918	1.4918	1.4934
0.5	1.6487	1.6487	1.6488	1.6504	1.6488	1.6488	1.6505	1.6487	1.6487	1.6504
0.6	1.8221	1.8221	1.8222	1.8237	1.8222	1.8222	1.8238	1.8221	1.8221	1.8237
0.7	2.0138	2.0138	2.0139	2.0150	2.0138	2.0139	2.0150	2.0138	2.0138	2.0150
0.8	2.2255	2.2255	2.2257	2.2263	2.2256	2.2256	2.2263	2.2255	2.2255	2.2263
0.9	2.4596	2.4596	2.4597	2.4596	2.4596	2.4597	2.4596	2.4596	2.4596	2.4597
1	2.7183	2.7182	2.7183	2.7174	2.7182	2.7183	2.7173	2.7182	2.7182	2.7175
L.S.E.		7.2959 e-9	6.804e-8	1.446e-5	1.413e-8	5.879e-8	1.66e-5	7.2951 e-9	9.338e-9	1.413e-5