Numerical Solution of Delay Differential Equations Using Legendre Wavelet Method

M. Sadeghi Hafshejani, S. Karimi Vanani and J. Sedighi Hafshejani

Department of Electrical Engineering, Islamic Azad University, Boroujen Branch P.O. Box 88715/141, Boroujen, Iran

Abstract: The purpose of this paper is to solve delay differential equations (DDEs) using Legendre wavelet method (LWM). The orthonormality of the basis functions using in this method is the main characteristic behind it to decrease the volume of computations and runtime of its algorithm. We state some concepts, properties and advantages of LWM and its applications for solving DDEs. Some illustrative numerical experiments including linear and nonlinear DDEs are given and some comparisons are made between LWM and variational iteration method [1], Adomian decomposition method [2] and homotopy perturbation method [3] to illustrate the validity and efficiency of the proposed method.

Key words: Delay differential equations • Legendre wavelet method

2010 Mathematics Subject Classification: Primary 34Kxx • Secondary 65T60

INTRODUCTION

In variant mathematical modelings, delay differential equation (DDEs) plays a crucial role in solving different sorts of problems. Moreover, it enjoys lots of applications in real-life problems such as infectious diseases, population dynamics, physiological and pharmaceutical kinetics, the navigational control of ships and aircrafts and control problems can be mentioned. For survey the reader is referred to [4] and [5]. There are many books in the field of DDEs and their applications. For instance, we can point out to the books of Driver [4], Gopalsamy [5], Halanay [6] and Kuang [7].

There are several numerical methods for solving DDEs; the important ones are the Bellman's method of steps [8], waveform relaxation method [9], Runge-Kutta method [10, 11], spline methods [12, 13], Radau IIA method [14], Multiquadric approximation scheme [15-19], variational iteration method (VIM) [1], Adomian decomposition method (ADM) [2] and homotopy perturbation method (HPM) [3].

In this paper, we are interested in solving DDEs using LWM as a powerful method. Wavelets theory is a relatively new and an emerging area in mathematical research. It has been applied in a wide range of engineering disciplines; particularly, wavelets are very successfully used in signal analysis for waveform representation and segmentations, time–frequency analysis and fast algorithms for easy implementation [20]. Wavelets permit the accurate representation of a variety of functions and operators. Moreover, wavelets establish a connection with fast numerical algorithms [21]. The application of Legendre wavelet for solving differential and integral equations is thoroughly considered in [22-27]. In this paper, the application of LWM for solving DDEs is considered and the properties of Legendre wavelets are utilized to evaluate an approximate solution for the desired DDE.

The paper is organized as follows. Section 2 gives some notations and basic definitions of the Legendre wavelets. Section 3 is devoted to the solution of DDEs using the LWM. In Section 4, we present some examples therein their numerical results demonstrate the high accuracy and efficiency of the proposed method. The discussion and conclusion are presented in Section 5.

Legendre Wavelet Method: Wavelets constitute a family of functions constructed from dilation and translation of a single function called the mother wavelet. When the dilation parameter and the translation parameter vary continuously, we have the following family of continuous wavelets as [28].
Where $a$ is dealation parameter and $b$ is translation parameter. If we restrict the parameters $a$ and $b$ to discrete values as $a = a_0^{-k}, b = nb_0$, $a_0 > 1, b_0 > 0$ and $n,k \in N$, we have the following family of discrete wavelets,

$$
\psi_{k,n}(t) = a_0^{k/2} \psi(a_0^k t - nb_0),
$$

Where $\psi_{k,n}(t)$ form a wavelet basis for $L^2(R)$. In particular, when $a_0 = 2$ and $b_0 = 1$ then $\psi_{k,n}(t)$ forms an orthonormal basis [29]. Legendre wavelet $\psi_{n,m}(t) = \psi(k,\hat{a},m,t)$ has four arguments; $\hat{a} = 2^{m-n-1}, n = 1,2,3,...,2^{k-1},k$ can assume any positive integer, $m$ is the order for Legendre polynomials and $t$ is the normalized time. They are defined on the interval [0, 1] as [29],

$$
\psi_{n,m}(t) = \begin{cases} 
\sqrt{\frac{1}{2^m}} P_m \left( 2^m t - \frac{\hat{a}}{2^m} \right), \\
0,
\end{cases}
$$

Where $m = 0,1,...,M-1, n = 1,2,3,...,2^{k-1}$. The coefficient $\sqrt{\frac{1}{2^m}}$ is for orthonormality, the dilation parameter is $a = 2^k$ and translation parameter is $b = \frac{\hat{a}}{2^m}$. Here, $P_{a}(t)$ are the well-known Legendre polynomials of order $m$ which are defined on the interval [1–11] and can be determined with the aid of the following recurrence formulae,

$$
P_0(t) = 1, P_1(t) = t,
$$

$$
P_{m+1}(t) = \left( \frac{2m+1}{m+1} \right) tP_m(t) - \left( \frac{m}{m+1} \right) P_{m-1}(t), m = 1,2,...
$$

A function $f(t)$ defined over [0, 1] may be expanded as

$$
f(t) = \sum_{n=0}^{\infty} \sum_{m=0}^{M-1} c_{nm} \psi_{n,m}(t),
$$

Where $c_{nm} = <f(t), \psi_{n,m}(t)>$, in which $<.,.>$ denotes the inner product. If the infinite series in equation (1) is truncated, then it can be written as

$$
y(t) \equiv \sum_{n=1}^{\hat{a}^{k-1}} \sum_{m=0}^{M-1} c_{nm} \psi_{n,m}(t) = C^T \psi(t),
$$

Where $C$ and $\psi(t)$ are $2^{k-1} M \times 1$ matrices given by

$$
C = \begin{bmatrix} c_{1,0} & \cdots & c_{1,M-1} & c_{2,0} & \cdots & c_{2,M-1} & \cdots & c_{2^{k-1},0} & \cdots & c_{2^{k-1},M-1} \end{bmatrix}^T,
$$

$$
\psi(t) = \begin{bmatrix} \psi_{1,0}(t) & \cdots & \psi_{1,M-1}(t) & \psi_{2,0}(t) & \cdots & \psi_{2,M-1}(t) & \cdots & \psi_{2^{k-1},0}(t) & \cdots & \psi_{2^{k-1},M-1}(t) \end{bmatrix}^T.
$$

**Application:** Let us consider a general form of DDEs with multiple delays as follows [11]:

$$
\frac{d^n y(t)}{dt^n} + \alpha(t) y(t) + \sum_{i=1}^{d} \beta_i(t) y(t-\tau_i) = 0,
$$

Where $n \geq 1$, $\alpha(t) \geq 0$, $\beta_i(t) \geq 0$, and $\tau_i \geq 0$ for $i = 1,2,...,d$. The solution of this equation can be obtained by using the wavelet techniques as described above.

\[ \begin{aligned}
    y'(t) &= f(t, y(t), y(\alpha_1(t)), \ldots, y(\alpha_r(t))), \quad t_0 \leq t \leq t_f, \\
    y(t) &= \phi(t), \quad t \leq t_0,
\end{aligned} \]  

(5)

Where \( f : [t_0, \infty) \times R^r \to R \) is a smooth function and \( \alpha_i(t), i = 1, 2, \ldots, r, \) are continuous functions on \([t_0, t_f] \times R\) such that \( \alpha_i(t) \leq t \). Also \( \phi(t) \) represents the initial function or the initial data. The aforementioned equation is one of the main delay differential equations including the following important pantograph equations,

\[ \begin{aligned}
    y'(t) &= ay(t) + by(rt), \quad 0 \leq t \leq t_f, \\
    y(0) &= y_0,
\end{aligned} \]

(6)

Where \( a, b \in C \) and \( 0 < r < 1 \). This equation is a very special delay differential equation that arises in quite different fields of pure and applied mathematics such as number theory, dynamical systems, probability, quantum mechanics and electrodynamics [1, 7, 31-36]. In particular, it was used by Okendon and Taylor [30] to study how to electric current collected by the pantograph of an electric locomotive.

In order to solve equation (5), we suppose that \([t_0, t_f] = [0, 1]\), otherwise the problem can be mapped from \([t_0, t_f]\) to \([0, 1]\) easily. For solving equation (5), it is sufficient to suppose that the approximate solution is as

\[ y(t) = C^\top \Psi(t), \]

Where \( C \) and \( \Psi(t) \) are given by equations (3) and (4). Therefore, we obtain

\[ \begin{aligned}
    C^\top (\Psi(t))' &= f(t, C^\top \Psi(t), C^\top \Psi(\alpha_1(t)), \ldots, C^\top \Psi(\alpha_r(t))), \quad t_0 \leq t \leq t_f, \\
    C^\top \Psi(t_0) &= \phi(t_0).
\end{aligned} \]

(6)

Therefore, in order to apply the LWM; we require \( 2^{k-1}M \) collocating points. Suitable collocating points are as [37]

\[ t_i = \cos \left( \frac{(2i + 1)\pi}{2^k M} \right), \quad i = 1, 2, \ldots, 2^{k-1}M. \]

Implementing the collocating points and imposing the initial value to the system (6), we obtain

\[ \begin{aligned}
    C^\top (\Psi(t_i))' &= f(t_i, C^\top \Psi(t_i), C^\top \Psi(\alpha_i(t_i)), \ldots, C^\top \Psi(\alpha_r(t_i))), i = 1, 2, \ldots, 2^{k-1}M - 1, \\
    C^\top \Psi(t_0) &= \phi(t_0).
\end{aligned} \]

The differential equation yields \( 2^{k-1}M - 1 \) equations and initial condition produces an equation. Therefore, the obtained system has \( 2^{k-1}M \) equations and \( 2^{k-1}M \) unknowns. Solving this system gives the unknown coefficients \( C \). We have used the Gauss elimination method with total pivoting to solve such a system.

**Numerical Experiments:** In this section, four experiments of DDEs are given and some comparisons are made to illustrate the efficiency of the method. The computations associated with the experiments discussed above were performed in Maple 14 on a PC with a CPU of 2.4 Ghz.
Experiment 4.1: Consider the following DDE [1-3, 12, 13, 18],

\[ y'(t) = \frac{1}{2} e^t y \left( \frac{t}{2} \right) + \frac{1}{2} y(t), \quad 0 < t \leq 1, \]

\[ y(0) = 1. \]

The exact solution is \( y(t) = e^t \).

We have solved this experiment using LWM with \( M = 13 \) and \( k = 1 \) and have compared it with thirteen terms of approximate solution obtained by several other methods. The comparison is given in Table 1.

The results in Table 1, illustrate that LWM is more accurate and efficient than spline methods, VIM, ADM and HPM. Hence, we prefer using LWM to solve the DDEs. Also, we find out the runtime of its algorithm is 0.171 in seconds. Therefore we can conclude that the algorithm of LWM is fast.

Experiment 4.2: Consider the following pantograph equation [11, 15],

\[ y'(t) - y \left( \frac{t}{2} \right) = 0, \quad 0 < t \leq 1, \]

\[ y(0) = 1. \]

The exact solution is

\[ y(t) = \sum_{k=0}^{\infty} \frac{1}{2^k} \frac{1}{k!} t^k. \]

We have solved this experiment using LWM with \( M = 9 \) and \( k = 2 \). The results are given in Table 2.

From the numerical results in Table 2, it is easy to conclude that obtained results using LWM are in good agreement with the exact solution. The runtime of the algorithm of the method is 0.091 in seconds.

<table>
<thead>
<tr>
<th>( t )</th>
<th>( u_{LWM} )</th>
<th>( u_{SM} )</th>
<th>( u_{VIM} )</th>
<th>( u_{ADM} )</th>
<th>( u_{HOM} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>0.2</td>
<td>1.37\times10^{-1}</td>
<td>0.00</td>
<td>3.10\times10^{-15}</td>
<td>7.54\times10^{-15}</td>
<td>0.00</td>
</tr>
<tr>
<td>0.4</td>
<td>3.27\times10^{-1}</td>
<td>1.00\times10^{-15}</td>
<td>1.39\times10^{-14}</td>
<td>2.00\times10^{-15}</td>
<td>5.00\times10^{-15}</td>
</tr>
<tr>
<td>0.6</td>
<td>5.86\times10^{-1}</td>
<td>2.19\times10^{-15}</td>
<td>2.13\times10^{-14}</td>
<td>5.00\times10^{-15}</td>
<td>1.00\times10^{-15}</td>
</tr>
<tr>
<td>0.8</td>
<td>9.54\times10^{-1}</td>
<td>9.36\times10^{-15}</td>
<td>3.19\times10^{-14}</td>
<td>5.00\times10^{-15}</td>
<td>1.00\times10^{-15}</td>
</tr>
<tr>
<td>1.0</td>
<td>1.43\times10^{-1}</td>
<td>1.72\times10^{-15}</td>
<td>3.19\times10^{-14}</td>
<td>5.00\times10^{-15}</td>
<td>1.00\times10^{-15}</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( t )</th>
<th>( u_{LWM} )</th>
<th>( u_{LWM} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>0.99999999999999</td>
<td>1.00000000000000</td>
</tr>
<tr>
<td>0.125</td>
<td>1.12894709929840</td>
<td>1.12894709929840</td>
</tr>
<tr>
<td>0.250</td>
<td>1.2659307192248</td>
<td>1.2659307192248</td>
</tr>
<tr>
<td>0.375</td>
<td>1.41126781788344</td>
<td>1.41126781788344</td>
</tr>
<tr>
<td>0.500</td>
<td>1.56514511174699</td>
<td>1.56514511174699</td>
</tr>
<tr>
<td>0.625</td>
<td>1.72784263272750</td>
<td>1.72784263272750</td>
</tr>
<tr>
<td>0.750</td>
<td>1.89962199489918</td>
<td>1.89962199489918</td>
</tr>
<tr>
<td>0.875</td>
<td>2.08074877752466</td>
<td>2.08074877752466</td>
</tr>
<tr>
<td>1.000</td>
<td>2.27149255550106</td>
<td>2.27149255550106</td>
</tr>
</tbody>
</table>
Table 3: Comparison between LWM, ADM, VIM, HPM and exact solution of Experiment 4.3

<table>
<thead>
<tr>
<th>t</th>
<th>( u_{ADM} )</th>
<th>( u_{VIM} )</th>
<th>( u_{HPM} )</th>
<th>( u_{Exact} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>0.0000000000</td>
<td>0.0000000000</td>
<td>0.0000000000</td>
<td>0.0000000000</td>
</tr>
<tr>
<td>0.125</td>
<td>0.124674731</td>
<td>0.124674731</td>
<td>0.124674733</td>
<td>0.124674733</td>
</tr>
<tr>
<td>0.250</td>
<td>0.247403957</td>
<td>0.247403957</td>
<td>0.247403959</td>
<td>0.247403959</td>
</tr>
<tr>
<td>0.375</td>
<td>0.366272528</td>
<td>0.366272527</td>
<td>0.366272529</td>
<td>0.366272529</td>
</tr>
<tr>
<td>0.500</td>
<td>0.479425533</td>
<td>0.479425537</td>
<td>0.479425538</td>
<td>0.479425538</td>
</tr>
<tr>
<td>0.625</td>
<td>0.585097232</td>
<td>0.585097271</td>
<td>0.585097272</td>
<td>0.585097272</td>
</tr>
<tr>
<td>0.750</td>
<td>0.681638554</td>
<td>0.681638759</td>
<td>0.681638760</td>
<td>0.681638760</td>
</tr>
<tr>
<td>0.875</td>
<td>0.767542679</td>
<td>0.767543501</td>
<td>0.767543502</td>
<td>0.767543502</td>
</tr>
<tr>
<td>1.000</td>
<td>0.841468253</td>
<td>0.841470984</td>
<td>0.841470984</td>
<td>0.841470984</td>
</tr>
</tbody>
</table>

Table 4: Approximate solution and exact solution of experiment 4.4

<table>
<thead>
<tr>
<th>t</th>
<th>( u_{LWM} )</th>
<th>( u_{Exact} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.000000000</td>
<td>0.000000000</td>
</tr>
<tr>
<td>0.1</td>
<td>0.09983342</td>
<td>0.09983341</td>
</tr>
<tr>
<td>0.2</td>
<td>0.19866933</td>
<td>0.19866933</td>
</tr>
<tr>
<td>0.3</td>
<td>0.29552020</td>
<td>0.29552020</td>
</tr>
<tr>
<td>0.4</td>
<td>0.38941834</td>
<td>0.38941834</td>
</tr>
<tr>
<td>0.5</td>
<td>0.47942553</td>
<td>0.47942553</td>
</tr>
<tr>
<td>0.6</td>
<td>0.56464247</td>
<td>0.56464247</td>
</tr>
<tr>
<td>0.7</td>
<td>0.64421768</td>
<td>0.64421768</td>
</tr>
<tr>
<td>0.8</td>
<td>0.71735609</td>
<td>0.71735609</td>
</tr>
<tr>
<td>0.9</td>
<td>0.78332691</td>
<td>0.78332690</td>
</tr>
<tr>
<td>1.0</td>
<td>0.84147098</td>
<td>0.84147098</td>
</tr>
</tbody>
</table>

Experiment 4.3: Consider the following nonlinear DDE [2, 18, 31],

\[ y'(t) = 1 - 2y(t)^2, \quad 0 < t \leq 1, \]
\[ y(0) = 0. \]

The exact solution is \( y(t) = \sin(t) \)

We have solved this experiment using LWM with \( M = 9 \) and \( k = 2 \) and have compared it with nine terms of approximate solution obtained by ADM, VIM and HPM. The comparison is given in Table 3.

A closer look at the results of the LWM in Table 3 reveals the higher-order accuracy of the proposed method among ADM, HPM and VIM. The runtime of the algorithm of the method is 1.420 in seconds.

Experiment 4.4: Consider the following nonlinear DDE,

\[ y'(t) + \sqrt{y(t)} + y(y(t)) + y(y^2(t)) = \cos(t) + \sqrt{\sin(t)} + \sin(\sin(t)) + \sin(\sin^2(t)), \quad 0 \leq t \leq 1, \]
\[ y(t) = \sin(t), \quad t \leq 0. \]

The exact solution is \( y(t) = \sin(t) \).

We have solved this experiment using LWM with \( M = 7 \) and \( k = 2 \). The obtained results are given in Table 4.

The results in Table 4, show that the obtained results using LWM are in good agreement with the exact solution. The runtime of the algorithm of the method is 3.307 in seconds.
CONCLUSIONS

In this study, the Legendre wavelet method is proposed for solving delay differential equations. Due to orthonormality of Legendre wavelets this method of solution is easy to implement and yields desired accuracy only in a few terms. As we observed, the method works excellently for nonlinear DDEs, too. This exhibits one of the other good advantages of the method in spite of its simplicity. The aforementioned advantages are shown that the LWM is a validate, reliable and promising tool.

REFERENCES