Research Article

Approximation Algorithm for a System of Pantograph Equations

Sabir Widatalla$^{1,2}$ and Mohammed Abdulai Koroma$^1$

$^1$ Department of Mathematics, Harbin Institute of Technology, Harbin 150001, China
$^2$ Department of Physics and Mathematics, College of Education, Sinnar University, Sinnar, Sudan

Correspondence should be addressed to Sabir Widatalla, sabirtag@yahoo.com

Received 23 November 2011; Accepted 2 January 2012

Academic Editor: Yuantong Gu

Copyright © 2012 S. Widatalla and M. A. Koroma. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We show how to adapt an efficient numerical algorithm to obtain an approximate solution of a system of pantograph equations. This algorithm is based on a combination of Laplace transform and Adomian decomposition method. Numerical examples reveal that the method is quite accurate and efficient, it approximates the solution to a very high degree of accuracy after a few iterates.

1. Introduction

The pantograph equation:

$$u'(t) = f(t, u(t), u(qt)), \quad t \geq 0,$$
$$u(0) = u_0,$$  \hspace{2cm} (1.1)

where $0 < q < 1$ is one of the most important kinds of delay differential equation that arise in many scientific models such as population studies, number theory, dynamical systems, and electrodynamics, among other. In particular, it was used by Ockendon and Tayler [1] to study how the electric current is collected by the pantograph of an electric locomotive, from where it gets its name.

The primary aim of this paper is to develop the Laplace decomposition for a system of multipantograph equations:
\begin{align}
  u'_1(t) &= \beta_1 u_1(t) + f_1(t, u_1(t), u_1(q_j)), \\
  u'_2(t) &= \beta_2 u_2(t) + f_2(t, u_1(t), u_1(q_j)), \\
  &\vdots \\
  u'_n(t) &= \beta_n u_n(t) + f_n(t, u_1(t), u_1(q_j)), \\
  u_i(0) &= u_{i0}, \quad i = 1, \ldots, n, \quad j = 1, 2, \ldots
\end{align}

where $\beta_i, u_{i0} \in \mathbb{C}$, and $f_i$ are analytical functions, and $0 < q_j < 1$.

In 2001, the Laplace decomposition algorithm (LDA) was proposed by Khuri in [2], who applied the scheme to a class of nonlinear differential equations. In this method, the solution is given as an infinite series usually converging very rapidly to the exact solution of the problem.

A major advantage of this method is that it is free from round-off errors and without any discretization or restrictive assumptions. Therefore, results obtained by LDA are more accurate and efficient. LDA has been shown to easily and accurately to approximate a solutions of a large class of linear and nonlinear ODEs and PDEs [2–4]. Ogun [5], for example, employed LDA to give an approximate solution of nonlinear ordinary Volterra integro-differential equations, Khan and Faraz [7] modified LDA to obtain series solutions of the boundary layer equation, and Yusufoglu [8] adapted LDA to solve Duffing equation.

The numerical technique of LDA basically illustrates how Laplace transforms are used to approximate the solution of the nonlinear differential equations by manipulating the decomposition method that was first introduced by Adomian [9, 10].

\section{2. Adaptation of Laplace Decomposition Algorithm}

We illustrate the basic idea of the Laplace decomposition algorithm by considering the following system:

\begin{align}
  L_i u_1 &= R_1(u_1, \ldots, u_n) + N_1(u_1, \ldots, u_n) + g_1, \\
  L_i u_2 &= R_2(u_1, \ldots, u_n) + N_2(u_1, \ldots, u_n) + g_2, \\
  &\vdots \\
  L_i u_n &= R_n(u_1, \ldots, u_n) + N_n(u_1, \ldots, u_n) + g_n.
\end{align}

With the initial condition

\begin{equation}
  u_i(0) = u_{i0}, \quad i = 1, \ldots, n
\end{equation}

where $L_i$ is first-order differential operator, $R_i$ and $N_i$, $i = 1, \ldots, n$, are linear and nonlinear operators, respectively, and $g_i$, $i = 1, \ldots, n$, are analytical functions.
The technique consists first of applying Laplace transform (denoted throughout this paper by $\mathcal{L}$) to the system of equations in (2.1) to get

\[
\begin{align*}
\mathcal{L}[L_1u_1] &= \mathcal{L}[R_1(u_1, \ldots, u_n)] + \mathcal{L}[N_1(u_1, \ldots, u_n)] + \mathcal{L}[g_1], \\
\mathcal{L}[L_1u_2] &= \mathcal{L}[R_2(u_1, \ldots, u_n)] + \mathcal{L}[N_2(u_1, \ldots, u_n)] + \mathcal{L}[g_2], \\
&\vdots \\
\mathcal{L}[L_1u_n] &= \mathcal{L}[R_n(u_1, \ldots, u_n)] + \mathcal{L}[N_n(u_1, \ldots, u_n)] + \mathcal{L}[g_n].
\end{align*}
\]

Using the properties of Laplace transform, and the initial conditions in (2.2) to get

\[
\begin{align*}
\mathcal{L}[u_1] &= \mathcal{H}_1(s) + \frac{1}{s}[\mathcal{L}[R_1(u_1, \ldots, u_n)] + \frac{1}{s}\mathcal{L}[N_1(u_1, \ldots, u_n)]], \\
\mathcal{L}[u_2] &= \mathcal{H}_2(s) + \frac{1}{s}[\mathcal{L}[R_2(u_1, \ldots, u_n)] + \frac{1}{s}\mathcal{L}[N_2(u_1, \ldots, u_n)]], \\
&\vdots \\
\mathcal{L}[u_n] &= \mathcal{H}_n(s) + \frac{1}{s}[\mathcal{L}[R_n(u_1, \ldots, u_n)] + \frac{1}{s}\mathcal{L}[N_n(u_1, \ldots, u_n)]],
\end{align*}
\]

where

\[
\mathcal{H}_i(s) = \frac{1}{s} (u_i(0) + \mathcal{L}[g_i]), \quad i = 1, \ldots, n.
\]

The Laplace decomposition algorithm admits a solution of $u_i(t)$ [2] in the form

\[
u_i(t) = \sum_{j=0}^{\infty} u_{ij}(t), \quad i = 1, \ldots, n,
\]

where the terms $u_{ij}(t)$ are to be recursively computed. The nonlinear operator $N_i$ is decomposed as follows:

\[
N_i(u_1, \ldots, u_n) = \sum_{j=0}^{\infty} A_{ij}.
\]
and $A_{ij}$ are the so-called Adomian polynomials that can be derived for various classes of non-linearity according to specific algorithms set by Adomian [9, 10].

\[ A_{i0} = f(u_{i0}), \]
\[ A_{i1} = u_{i1} f'(u_{i0}), \]
\[ A_{i2} = u_{i2} f'(u_{i0}) + \frac{1}{2!} u_{i1}^2 f''(u_{i0}), \]
\[ A_{i3} = u_{i3} f'(u_{i0}) + u_{i1} u_{i2} f''(u_{i0}) + \frac{1}{3!} u_{i1}^3 f'''(u_{i0}), \ldots \]  

(2.8)

Substituting (2.6) and (2.7) into (2.4), and using the linearity of Laplace transform, we get

\[
\sum_{j=0}^\infty \mathcal{L}[u_{ij}] = \mathcal{H}_1(s) + \frac{1}{s} \sum_{j=0}^\infty \mathcal{L}[R_1(u_{ij}, \ldots, u_{nj})] + \frac{1}{s} \sum_{j=0}^\infty \mathcal{L}[A_{ij}],
\]
\[
\sum_{j=0}^\infty \mathcal{L}[u_{i2}] = \mathcal{H}_2(s) + \frac{1}{s} \sum_{j=0}^\infty \mathcal{L}[R_2(u_{ij}, \ldots, u_{nj})] + \frac{1}{s} \sum_{j=0}^\infty \mathcal{L}[A_{ij}],
\]
\[
\vdots
\]
\[
\sum_{j=0}^\infty \mathcal{L}[u_{ijn}] = \mathcal{H}_n(s) + \frac{1}{s} \sum_{j=0}^\infty \mathcal{L}[R_n(u_{ij}, \ldots, u_{nj})] + \frac{1}{s} \sum_{j=0}^\infty \mathcal{L}[A_{ijn}].
\]  

(2.9)

We thus have the following recurrence relations from corresponding terms on both sides of (2.9):

\[ \mathcal{L}[u_{i0}(t)] = \mathcal{H}_i(s), \]  

(2.10)

\[ \mathcal{L}[u_{i1}(t)] = \frac{1}{s} \mathcal{L}[R(u_{i1}, \ldots, u_{n0})] + \frac{1}{s} \mathcal{L}[A_{i0}], \]  

(2.11)

\[ \mathcal{L}[u_{i2}(t)] = \frac{1}{s} \mathcal{L}[R(u_{i1}, \ldots, u_{n1})] + \frac{1}{s} \mathcal{L}[A_{i1}], \ldots \]  

(2.12)

Generally,

\[ \mathcal{L}[u_{i(j+1)}(t)] = \frac{1}{s} \mathcal{L}[R(u_{i1}, \ldots, u_{nj})] + \frac{1}{s} \mathcal{L}[A_{ij}]. \]  

(2.13)

Applying the inverse Laplace transform to (2.10) gives the initial approximation

\[ u_{i0}(t) = \mathcal{L}^{-1}[\mathcal{H}_i(s)], \quad i = 1, \ldots, n. \]  

(2.14)
Substituting these values of $u_{i0}$ into the inverse Laplace transform of (2.11) gives $u_{i1}$. The other terms $u_{i2}, u_{i3}, \ldots$ can be obtained recursively in similar fashion from

$$u_{i(j+1)}(t) = \mathcal{L}^{-1} \left[ \frac{1}{s} \mathcal{L}[R(u_{ij}, \ldots, u_{nj})] + \frac{1}{s} \mathcal{L}[A_{ij}] \right], \quad j = 0, 1, 2, \ldots \quad (2.15)$$

To provide clearly a view of the analysis presented above, three illustrative systems of pantograph equations have been used to show the efficiency of this method.

3. Test Problems

All iterates are calculated by using Matlab 7. The absolute errors in Tables 1–3 are the values of $|u_i(t) - \sum_{j=0}^{n} u_{ij}(t)|$, those at selected points.

Example 3.1. Consider the two-dimensional pantograph equations:

\begin{align*}
  u'_1 &= u_1(t) - u_2(t) + u_1 \left( \frac{t}{2} \right) - e^{t/2} + e^{-t}, \\
  u'_2 &= -u_1(t) - u_2(t) - u_2 \left( \frac{t}{2} \right) + e^{t/2} + e^t \\
  u_1(0) &= 1, \quad u_2(0) = 1.
\end{align*}

(3.1)

Applying the result of (2.14) gives us

\begin{align*}
  u_{10}(t) &= 4 - 2e^{t/2} - e^{-t}, \\
  u_{20}(t) &= 2 - 2e^{-t/2} + e^t. \quad (3.2)
\end{align*}

The iteration formula (2.15) for this example is

\begin{align*}
  u_{1(j+1)} &= \mathcal{L}^{-1} \left[ \frac{1}{s} \mathcal{L} \left( u_{1j}(t) - u_{2j}(t) + u_{1j} \left( \frac{t}{2} \right) \right) \right], \\
  u_{2(j+1)} &= \mathcal{L}^{-1} \left[ \frac{1}{s} \mathcal{L} \left( -u_{1j}(t) - u_{2j}(t) - u_{2j} \left( \frac{t}{2} \right) \right) \right]. \quad (3.3)
\end{align*}
Starting with an initial approximations $u_{10}(t)$ and $u_{20}(t)$ and use the iteration formula (3.3). We can obtain directly the other components as

$$u_{11}(t) = 14 + 6t + e^{-t} - e^t - 2e^{-t}/2 - 4e^{t/2} - 8e^{t/4},$$

$$u_{21}(t) = 12 - 8t - e^{-t} - e^t - 4e^{-t}/2 + 2e^{t/2} - 8e^{-t}/4,$$

$$u_{12}(t) = 158 + 16t + \frac{17}{2}t^2 - 2e^{-t} - 14e^{t/2} - 6e^{-t/2} - 48e^{t/4} - 24e^{-t/4} - 64e^{t/8},$$

$$u_{22}(t) = 158 + 16t + \frac{17}{2}t^2 - 2e^{-t} - 14e^{t/2} - 6e^{-t/2} - 48e^{t/4} - 24e^{-t/4} - 64e^{t/8}. \tag{3.4}$$

Table 1 shows the absolute error of LDA with $n = 2, 4,$ and $6.$

**Example 3.2.** Consider the system of multipantograph equations:

$$u'_1(t) = -u_1(t) - e^{-t} \cos \left( \frac{t}{2} \right) u_2 \left( \frac{t}{2} \right) - 2e^{-2(t/4)} \cos \left( \frac{t}{2} \right) \sin \left( \frac{t}{4} \right) u_1 \left( \frac{t}{4} \right),$$

$$u'_2(t) = e^t u_2^2 \left( \frac{t}{2} \right) - u_2^2 \left( \frac{t}{2} \right),$$

$$u_1(0) = 1, \quad u_2(0) = 0. \tag{3.5}$$

Let us start with an initial approximation:

$$u_{10}(t) = 1,$$

$$u_{20}(t) = 0. \tag{3.6}$$
The iteration formula

\[ u_{1(i+1)} = \mathcal{L}^{-1}\left[ \frac{1}{s} \mathcal{L}\left( u_{1j}(t) - u_{2j}(t) + u_{1j}\left( \frac{t}{2} \right) \right) \right], \]

\[ u_{2(i+1)} = \mathcal{L}^{-1}\left[ \frac{1}{s} \mathcal{L}\left( e^{A \left( \frac{t}{2} \right)} - A_{2j} \right) \right], \] (3.7)

where

\[ A_{i0} = u_{i0}^2\left( \frac{t}{2} \right), \]

\[ A_{i1} = 2u_{i0}\left( \frac{t}{2} \right) u_{i1}\left( \frac{t}{2} \right), \]

\[ A_{i2} = u_{i1}^2\left( \frac{t}{2} \right) + 2u_{i0}\left( \frac{t}{2} \right) u_{i2}\left( \frac{t}{2} \right), \]

\[ A_{i3} = 2u_{i1}\left( \frac{t}{2} \right) u_{i2}\left( \frac{t}{2} \right) + 2u_{i0}\left( \frac{t}{2} \right) u_{i3}\left( \frac{t}{2} \right), \ldots, \quad i = 1, 2. \] (3.8)

Table 2 shows the absolute error of LDA with \( n = 1, 2, \) and 3.

Example 3.3. Consider the three-dimensional pantograph equations:

\[ u_1'(t) = 2u_2\left( \frac{t}{2} \right) + u_3(t) - t \cos\left( \frac{t}{2} \right), \]

\[ u_2'(t) = 1 - t \sin(t) - 2u_3\left( \frac{t}{2} \right), \]

\[ u_3'(t) = -2u_2\left( \frac{t}{2} \right) - u_3(t). \]
\[ u'_3(t) = u_2(t) - u_1(t) - t \cos(t), \]
\[ u_1(0) = -1, \quad u_2(0) = 0, \quad u_3(0) = 0. \quad (3.9) \]

By (2.14) our initial approximation is
\[ u_{10}(t) = 3 - 4 \cos\left(\frac{t}{2}\right) - 2t \sin\left(\frac{t}{2}\right), \]
\[ u_{20}(t) = -\sin(-t) + t \cos(t) + t, \quad (3.10) \]
\[ u_{30}(t) = -\cos(t) - t \sin(t) + 1. \]

The iteration formula (2.15) for this example is
\[
\begin{align*}
    u_{1(j+1)} &= \mathcal{L}^{-1} \left[ \frac{1}{s} \mathcal{L} \left( 2u_{2j} \left( \frac{t}{2} \right) + u_{3j}(t) \right) \right], \\
    u_{2(j+1)} &= \mathcal{L}^{-1} \left[ \frac{1}{s} \mathcal{L} \left( -2A_{2j} \right) \right], \\
    u_{3(j+1)} &= \mathcal{L}^{-1} \left[ \frac{1}{s} \mathcal{L} \left( u_{2j}(t) - u_{1j}(t) \right) \right],
\end{align*}
\]
where
\[
\begin{align*}
    A_{20} &= u_{20}^2 \left( \frac{t}{2} \right), \\
    A_{21} &= 2u_{20} \left( \frac{t}{2} \right) u_{21} \left( \frac{t}{2} \right), \\
    A_{22} &= u_{21}^2 \left( \frac{t}{2} \right) + 2u_{20} \left( \frac{t}{2} \right) u_{22} \left( \frac{t}{2} \right), \\
    A_{23} &= 2u_{21} \left( \frac{t}{2} \right) u_{22} \left( \frac{t}{2} \right) + 2u_{20} \left( \frac{t}{2} \right) u_{23} \left( \frac{t}{2} \right), \\
    \vdots
\end{align*}
\]

Table 3 shows the absolute error of LDA with \( n = 1, 2, \) and 3.

4. Conclusion

The main objective of this paper is to adapt Laplace decomposition algorithm to investigate systems of pantograph equations. We also aim to show the power of the LAD method by reducing the numerical calculation without need to any perturbations, discretization, or/and other restrictive assumptions which may change the structure of the problem being
solved. LDA method gives rapidly convergent successive approximations through the use of recurrence relations. We believe that the efficiency of the LDA gives it a much wider applicability.

References
