TÜBİTAK

Turkish Journal of Mathematics

http://journals.tubitak.gov.tr/math/

Research Article

Turk J Math
(2020) 44: 85 – 97
© TÜBİTAK
doi:10.3906/mat-1908-67

A Galerkin-like method for solving linear functional differential equations under initial conditions

Şuayip YÜZBAŞI*, **Murat KARAÇAYIR**
Department of Mathematics, Faculty of Science, Akdeniz University, Antalya, Turkey

Received: 20.08.2019 • **Accepted/Published Online:** 24.10.2019 • **Final Version:** 20.01.2020

Abstract: In this paper, we present a weighted residual Galerkin method to solve linear functional differential equations. We consider the problem with variable coefficients under initial conditions. Assuming the exact solution of the problem has a Taylor series expansion convergent in the relevant domain, we seek a solution of the given problem in the form of a polynomial having degree N of our choice. Substituting this polynomial with unknown coefficients in the given equation yields an expression linear in these coefficients. We then proceed as in the weighted residual method and take inner product of this expression with monomials up to degree N, resulting in $N+1$ linear algebraic equations. Appropriately incorporating the initial conditions and solving the resulting linear system, we obtain the approximate solution to the given problem. Additionally, we present a way of estimating the absolute error of the obtained approximation, which is then used to improve the original approximation through a method called residual correction. We also show that the upper bound for the error of the proposed method depends on the Taylor truncation error of the exact solution. The proposed scheme and the residual correction technique are illustrated in several example problems.

Key words: Functional differential equations, generalized pantograph equation, weighted residual method, method of moments, numerical solutions, residual error correction

1. Introduction

The first order delay differential equation (DDE) with constant coefficients and proportional delay

$$u'(x) = au(x) + bu(qx),\ x \in I = [0,T],\ 0 < q < 1 \tag{1.1}$$

is called the pantograph equation and first appeared in the mathematical modelling of the wave motion in the current line between an electric locomotive and its overhead catenary wire [1, 2]. For any $u_0 \in \mathbb{R}$, it has a unique (continuously differentiable) solution $u(x)$ over I satisfying the condition $u(0) = u_0$. More generally, if the coefficients a and b in equation (1.1) are replaced with functions from $C^m(I)$, then a unique solution $u(x)$ from $C^{m+1}(I)$ can be found which also satisfies the same initial condition [3]. interested reader is referred to [4, 5] for analytical and approximate solutions of the pantograph equation and for DDEs in general.

The pantograph equation plays a role in a wide range of applications. This has brought about an increasing interest on the numerical solution of DDEs of pantograph type and researchers have made use of numerous methods. To name a few, Adomian decomposition method was used for this purpose by Dehghan et al. in [6], where the convergence of the approach was also established. Another popular method, homotopy

*Correspondence: syuzbasi@akdeniz.edu.tr
2010 *AMS Mathematics Subject Classification:* 34K06, 65L60

perturbation method, was used by Feng [7], where the multipantograph equation was considered with variable coefficients. The authors of this paper used a weighted residual Galerkin approach [8] to obtain approximate solutions of the same problem. Ishiwata and Muroya used two kinds of collocation methods based on special mesh distributions in [9]. Other collocation methods were presented in conjunction with Bernstein [10] and Bessel [11] polynomials for the same purpose. In [12], pantograph type DDEs were solved by shifted Chebyshev polynomials and an applicable error analysis was presented. Another collocation method that is based on exponential functions rather than polynomials can be found in [13].

In this paper, we consider the linear nonhomogeneous m-th order generalized pantograph equation with variable coefficients, given by

$$y^{(m)}(x)+\sum_{j=0}^{J}\sum_{k=0}^{m-1}P_{jk}(x)y^{(k)}(\lambda_{jk}x+\mu_{jk})=f(x),\ 0\le x\le b, \tag{1.2}$$

under the initial conditions

$$\sum_{k=0}^{m-1}a_{ik}y^{(k)}(0)=\alpha_i,\ i=0,1,\ldots,m-1. \tag{1.3}$$

Here, the given functions $P_{jk}(x)$ and $f(x)$ in equation (1.2) are defined on the interval $[0,b]$ and the coefficients $\lambda_{jk},\mu_{jk},a_{ik},\alpha_i$ are real numbers, where the discrete delay elements μ_{jk} are typically negative. Theoretically, in equation (1.2), any number of combinations of proportional and discrete delays of x may correspond to derivatives of $y(x)$ of any order k. This means that no upper bound can be specified for the parameter J, although it does not usually exceed 1 in the problems considered in the literature. Contrary to the pantograph equation, the order of generalized pantograph equations can be greater than 1, and its terms contain both simple and proportional delay elements. Generalized pantograph equations are encountered in the applications that model various physical phenomena and a large variety of numerical models have been used to find approximate solutions of equations of this type. To name a few, these methods include a collocation algorithm based on Bernoulli polynomials [14], a Taylor polynomial approach [15], a collocation method involving Lucas polynomials [16], variational iteration method [17], and an operational matrix method using polynomials in the standard basis [18]. In addition to these, pantograph equations have been solved using shifted orthonormal Bernstein polynomials [19]. There are also studies interested in solving a nonlinear version of equation (1.2); the reader can see [20], for example, where Lucas polynomials are employed in conjunction with collocation points to solve nonlinear delay differential equations of this type.

The organization of the paper is as follows: The proposed solution method for equation (1.2) is explained in Section 2. In Section 3, firstly the technique of residual correction is described, which exploits the linearity of equation (1.2) in order to estimate the absolute error and thus obtain a more accurate approximation from the original one. Secondly, assuming the exact solution of equation (1.2) is analytic, an upper bound for the absolute error of the approximate solution is given in terms of the Taylor truncation error of the exact solution. Section 4 contains numerical examples, where approximate solutions corresponding to various N values are obtained using the proposed method. In addition, residual correction technique described in Section 3.1 is applied to two example problems and comparisons with other methods are made whenever possible. Finally, conclusions of the paper are summarized in Section 5.

2. Solution method

In this section, we outline the Galerkin-like procedure we will use in order to solve equation (1.2). The method, which relies on taking inner product of an expression with monomials, also called the method of moments, was employed to obtain approximate solutions of high-order Fredholm integro-differential equations [21] and high-order integro-differential equations with weakly singular kernel [22]. In this study, we develop the method to compute the approximate solutions of generalized pantograph-type functional differential equations.

To begin with, we assume that the unique solution $y(x)$ of equation (1.2) can be expressed as a power series of the form

$$y(x) = \sum_{k=0}^{\infty} a_k x^k.$$

We then truncate this power series after the $(N+1)$st term so that

$$y_N(x) = \sum_{k=0}^{N} a_k x^k = \mathbf{X}(x) \cdot \mathbf{A} \tag{2.1}$$

where

$$\mathbf{X}(x) = [\ 1 \quad x \quad x^2 \quad \dots \quad x^N\ ],\ \mathbf{A} = [\ a_0 \quad a_1 \quad a_2 \quad \dots \quad a_N\ ]^T.$$

Here, the coefficients a_i will be obtained as the output of the procedure. The derivative $y_N^{'}(x)$ of the approximate solution can also be expressed as a product of matrices. Namely, if $\mathbf{B}$ is the $(N+1) \times (N+1)$ matrix

$$\mathbf{B} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & N \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}$$

defined by $\mathbf{B}_{i,i+1} = i$ for $i = 1, 2, \dots, N$ and $\mathbf{B}_{i,j} = 0$ elsewhere, then the equality

$$y_N^{(k)}(x) = \mathbf{X}(x)\mathbf{B}^k\mathbf{A} \tag{2.2}$$

holds for any nonnegative integer k, where $y_n^{(0)}$ denotes the function y_n itself. In order to express the delayed terms in equation (1.2), equation (2.2) can be utilized in a straightforward manner. For this purpose, given two constants λ and μ, we replace the variable x in $\mathbf{X}(x)$ with $\lambda x + \mu$ and write equation (2.2) as

$$y_N^{(k)}(\lambda x + \mu) = \mathbf{X}(\lambda x + \mu)\mathbf{B}^k\mathbf{A}. \tag{2.3}$$

It is worth noting that the vector $\mathbf{X}(\lambda x + \mu)$ can be expressed in a form that is more convenient for computer implementations. Namely, if we define the $(N+1)$-dimensional square matrix $\mathbf{B}(\lambda, \mu)$ by $\mathbf{B}_{i,j}(\lambda, \mu) =$

$\binom{j-1}{i-1}\lambda^{i-1}\mu^{j-i}$ where we use the conventions $\binom{i}{j}=0$ whenever $j>i$ and $\binom{0}{0}=1$, or more explicitly by

$$\mathbf{B}(\lambda,\mu)=\begin{bmatrix} 1 & \mu & \mu^2 & \dots & \mu^N \\ 0 & \lambda & 2\lambda\mu & \dots & N\lambda\mu^{N-1} \\ 0 & 0 & \lambda^2 & \dots & \frac{N(N-1)}{2}\lambda^2\mu^{N-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \lambda^N \end{bmatrix},$$

then the equality $\mathbf{X}(\lambda x+\mu)=\mathbf{X}(x)\mathbf{B}(\lambda,\mu)$ holds.

After that, substituting the matrix expressions for $y_N^{(m)}(x)$ and $y_N^{(k)}(\lambda_{jk}x+\mu_{jk})$ given by equation (2.2) and (2.3) for $j=0,1,\dots,J$ and $k=0,1,\dots,m-1$ into equation (1.2) and rearranging yields

$$\left\{\mathbf{X}\mathbf{B}^m+\sum_{j=0}^{J}\sum_{k=0}^{m-1}P_{jk}(x)\mathbf{X}(x)\mathbf{B}(\lambda_{jk},\mu_{jk})\mathbf{B}^k\right\}\mathbf{A}=f(x). \tag{2.4}$$

Now, since we are basically trying to calculate the approximate solution $y_N(x)$ in terms of the elements of the basis set is $\mathbf{\Phi}=\{1,x,x^2,\dots,x^N\}$ of the set of all polynomials having degree at most N, the coefficients a_i can be determined by taking inner product of the elements of $\mathbf{\Phi}$ with the vector of length $N+1$ on the left-hand side of equation (2.4). More precisely, if we define $\mathbf{G}(x)$ by

$$\mathbf{G}(x):=\mathbf{X}\mathbf{B}^m+\sum_{j=0}^{J}\sum_{k=0}^{m-1}P_{jk}(x)\mathbf{X}(x)\mathbf{B}(\lambda_{jk},\mu_{jk})\mathbf{B}^k,$$

taking the inner product of each element of $\mathbf{\Phi}$ with both sides of equation (2.4) results in a linear system $\mathbf{WA}=\mathbf{F}$ of $N+1$ equations in the unknowns $a_0,a_1,\dots,a_N$. Explicitly, the coefficient matrix $\mathbf{W}$ and the column vector $\mathbf{F}$ on the right-hand side of this system is given by $\mathbf{W}_{i,j}=<x^{i-1},\mathbf{G}(x)_{1,j}>$, $\mathbf{F}_{i,1}=<x^{i-1},f(x)>$, or more explicitly by

$$\mathbf{W}=\begin{bmatrix} <1,\mathbf{G}(x)_{1,1}> & <1,\mathbf{G}(x)_{1,2}> & \dots & <1,\mathbf{G}(x)_{1,N+1}> \\ <x,\mathbf{G}(x)_{1,1}> & <x,\mathbf{G}(x)_{1,2}> & \dots & <x,\mathbf{G}(x)_{1,N+1}> \\ <x^2,\mathbf{G}(x)_{1,1}> & <x^2,\mathbf{G}(x)_{1,2}> & \dots & <x^2,\mathbf{G}(x)_{1,N+1}> \\ \vdots & \vdots & \vdots & \vdots \\ <x^N,\mathbf{G}(x)_{1,1}> & <x^N,\mathbf{G}(x)_{1,2}> & \dots & <x^N,\mathbf{G}(x)_{1,N+1}> \end{bmatrix}, \tag{2.5}$$

$$\mathbf{F}=\begin{bmatrix} <1,f(x)> & <x,f(x)> & <x^2,f(x)> & \dots & <x^N,f(x)> \end{bmatrix}^T,$$

where the inner product is defined by $<f,g>=\int_0^b f(x)g(x)\mathrm{d}x$. Here, f and g are functions from the Hilbert space $L^2[0,b]$. Since the initial conditions in equation (1.3) should also be satisfied, m rows of $\mathbf{W}$ and the corresponding m entries of $\mathbf{F}$ should be modified accordingly. For the sake of being deterministic, let us fix the modified entries to be the last m rows of $\mathbf{W}$ and the last m entries of $\mathbf{F}$. Determining the matrix equivalents of the initial conditions is easy in view of equation (2.2). Namely, $(i+1)$-st initial condition is given by

$$\sum_{k=0}^{m-1}a_{ik}y^{(k)}(0)=\sum_{k=0}^{m-1}[a_{ik}\mathbf{X}(0)]\mathbf{B}^k\mathbf{A}=\alpha_i$$

for $i = 0, 1, \ldots, m-1$, where $\mathbf{X}(0) = [\ 1 \quad 0 \quad \cdots \quad 0\ ]$. Thus, feeding the m initial conditions into the system $\mathbf{WA} = \mathbf{F}$ amounts to substituting the $m \times (N+1)$ matrix

$$\begin{bmatrix} \sum_{k=0}^{m-1}[a_{0k}\mathbf{X}(0)]\mathbf{B}^k \\ \sum_{k=0}^{m-1}[a_{1k}\mathbf{X}(0)]\mathbf{B}^k \\ \vdots \\ \sum_{k=0}^{m-1}[a_{m-1,k}\mathbf{X}(0)]\mathbf{B}^k \end{bmatrix}$$

for the last m rows of $\mathbf{W}$ and the length-m vector $[\ \alpha_0 \quad \alpha_1 \quad \ldots \quad \alpha_{m-1}\ ]^T$ for the last m entries of $\mathbf{F}$. After performing this step, we are left with the modified system $\tilde{\mathbf{W}}\mathbf{A} = \tilde{\mathbf{F}}$ explicitly given by

$$\tilde{\mathbf{W}} = \begin{bmatrix} < 1, \mathbf{G}(x)_{1,1} > & < 1, \mathbf{G}(x)_{1,2} > & \ldots & < 1, \mathbf{G}(x)_{1,N+1} > \\ < x, \mathbf{G}(x)_{1,1} > & < x, \mathbf{G}(x)_{1,2} > & \ldots & < x, \mathbf{G}(x)_{1,N+1} > \\ < x^2, \mathbf{G}(x)_{1,1} > & < x^2, \mathbf{G}(x)_{1,2} > & \ldots & < x^2, \mathbf{G}(x)_{1,N+1} > \\ \vdots & \vdots & \vdots & \vdots \\ < x^{N-m}, \mathbf{G}(x)_{1,1} > & < x^{N-m}, \mathbf{G}(x)_{1,2} > & \ldots & < x^{N-m}, \mathbf{G}(x)_{1,N+1} > \\ \sum_{k=0}^{m-1}[a_{0k}\mathbf{X}(0)]\mathbf{B}^k & & & \\ \sum_{k=0}^{m-1}[a_{1k}\mathbf{X}(0)]\mathbf{B}^k & & & \\ \vdots & & & \\ \sum_{k=0}^{m-1}[a_{m-1,k}\mathbf{X}(0)]\mathbf{B}^k & & & \end{bmatrix},$$

$$\tilde{\mathbf{F}} = [\ < 1, f(x) > \quad < x, f(x) > \quad \ldots \quad < x^{N-m}, f(x) > \quad \alpha_0 \quad \alpha_1 \quad \cdots \quad \alpha_{m-1}\ ]^T,$$

from which we obtain the matrix of unknown coefficients $\mathbf{A} = \tilde{\mathbf{W}}^{-1}\tilde{\mathbf{F}}$ and the approximate solution

$$y_N(x) = a_0 + a_1 x + \ldots + a_N x^N.$$

3. Error analysis and residual correction

In this section, we first present a way of improving the accuracy of an already obtained approximate solution. Then, we give an upper bound for the absolute error in terms of the Taylor truncation error of the exact solution.

3.1. Error estimation and residual correction

In situations where it is not possible to measure the accuracy of an approximate solution, the residual function is useful since it gives an idea about the efficiency of this approximation. Furthermore, when the equation is linear in the unknown function, it is possible to exploit the residual function of the approximate solution in order to obtain a more accurate solution. In our case, since the generalized pantograph equation given by equations (1.2) is linear in the unknown function $y(x)$, the method of residual correction can be applied in a rather straightforward manner. This section is about how the residual function gives rise to an error estimation of an approximate solution to equations (1.2) obtained as the output of our method. We also exhibit how this estimation is used to obtain better approximate solutions.

To this end, let us consider the residual function

$$R(x) = y^{(m)}(x) + \sum_{j=0}^{J}\sum_{k=0}^{m-1} P_{jk}(x) y^{(k)}(\lambda_{jk}x + \mu_{jk}) - f(x) = 0 \tag{3.1}$$

of equation (1.2). We now replace $y(x)$ by the approximate solution $y_N(x)$ to obtain

$$R_N(x) = y_N^{(m)}(x) + \sum_{j=0}^{J}\sum_{k=0}^{m-1} P_{jk}(x) y_N^{(k)}(\lambda_{jk}x + \mu_{jk}) - f(x) \tag{3.2}$$

as the residual function of $y_N(x)$. Subtracting equation (3.1) from equation (3.2) and rearranging yields

$$e_N^{(m)}(x) + \sum_{j=0}^{J}\sum_{k=0}^{m-1} P_{jk}(x) e_N^{(k)}(\lambda_{jk}x + \mu_{jk}) = -R_N(x), \tag{3.3}$$

which is just equation (1.2) with $y(x)$ replaced by $e_N(x) = y(x) - y_N(x)$ and with the nonhomogeneous term $R_N(x)$ instead of $f(x)$. Furthermore, since the approximate solution $y_N(x)$ also satisfies the initial conditions in equation (1.3), a simple manipulation gives

$$\sum_{k=0}^{m-1} a_{ik} e_N^{(k)}(0) = 0,\ i = 0, 1, \ldots, m-1 \tag{3.4}$$

as the initial conditions of equation (3.3). Next step is to simply employ the method described in Section 2 to find an approximate solution to equation (3.3) for some choice of positive integer M. This approximate solution, which we will denote by $e_{N,M}(x)$, is our estimation of the error function $e_N(x)$ for the approximate solution $y_N(x)$ to equation (1.2). Thus, our new approximate solution, called the corrected solution, to equation (1.2) is given by

$$y_{N,M}(x) = y_N(x) + e_{N,M}(x). \tag{3.5}$$

In what follows, we will denote by $E_{N,M}(x)$ the actual error of the corrected solution, which is equal to $E_{N,M}(x) = y(x) - y_{N,M}(x)$. Note that the relation $E_{N,M}(x) = e_N(x) - e_{N,M}(x)$ also holds between the estimated error $e_{N,M}(x)$ of $y_N(x)$ and its actual error $e_N(x)$. This means that we can measure the success of the residual correction process by the accuracy of our estimation of the error function $e_N(x)$ corresponding to $y_N(x)$. This will be made clear in the examples of the succeeding section.

3.2. Error bound for the solution

In this part, we relate the error bound for the approximate solution $y_N(x)$ to the truncation error of the Taylor polynomial corresponding to the exact solution.

Theorem 3.1 *Let $y_N(x)$ and $y(x)$ denote the approximate and the exact solutions of problem (1.2), respectively. If $y(x) \in C^{N+1}[0,b]$, then*

$$|y(x) - y_N(x)| \le |R_N^T(x)| + |y_N^T(x) - y_N(x)| \tag{3.6}$$

where $y_N^T(x)$ denotes the Nth degree Taylor polynomial of y around the point $x = q \in [0,b]$ and R_N^T represents its remainder term.

Proof Since y is $(N+1)$-times continuously differentiable, it can be represented by its Taylor series as

$$y(x) = \sum_{n=0}^{N} \frac{(x-q)^n}{n!} y^n(q) + R_N^T(x),$$

where

$$R_N^T(x) = \frac{(x-q)^{N+1}}{(N+1)!} y^{(N+1)}(c_x),\ 0 < c_x < x \leq b$$

is the reminder term of the Taylor expansion of y. Thus, $y(x) - y_N^T(x) = R_N^T(x)$. By using this and the triangle inequality, we obtain

$$\begin{aligned} |y(x) - y_N(x)| &= |y(x) - y_N(x) + y_N^T(x) - y_N^T(x)| \\ &\leq |y(x) - y_N^T(x)| + |y_N^T(x) - y_N(x)| \\ &= |R_N^T(x,q)| + |y_N^T(x) - y_N(x)|. \end{aligned}$$

□

As a result, we have found an upper bound of the absolute error in terms of the Taylor truncation error of the exact solution. Note that this is not an a priori error bound; it only serves as a means to compare the actual error to this Taylor truncation error.

4. Illustrative examples

In this section, we apply the method explained in Section 2 to several generalized pantograph equations and compare the resulting approximate solutions with some other methods present in the literature. We also apply residual correction to obtain better approximate solutions using the existing ones. All the calculations have been performed using MATLAB.

Example 4.1 *Let us first consider the third-order generalized pantograph equation studied in [23] and then in [14, 24, 25]:*

$$\begin{aligned} y'''(t) + y(t) + y(t-0.3) &= e^{-t+0.3},\ 0 \leq t \leq 1 \\ y(0) = 1, y'(0) &= -1, y''(0) = 1. \end{aligned} \tag{4.1}$$

The exact solution of this problem is $y(t) = e^{-t}$. Table 1 compares the absolute errors of the solutions obtained by the present method, the Taylor series method [23], the Hermite method [24], the Chebyshev-Gauss grid method [25], and the collocation method based on Bernoulli operational matrix [14] for $N = 5$ and $N = 8$. It can be concluded that for each choice of the parameter N, the present method outperforms the aforementioned ones for most, if not all, of the sample points taken from $[0,1]$. The values in the table also imply that the absolute error functions resulting from the present method is more evenly distributed over the interval $[0,1]$ compared to the other methods. Furthermore, the last column indicates that increasing N decreases the absolute error by a significant amount. The data in the table can be seen in a visual setting in Figure 1.

We have also applied residual correction to the approximate solution $y_8(x)$ as explained in Section 3.1 in this problem. For this purpose, firstly we have choosen $M = 10$ and $M = 13$ in order to obtain the estimates for the actual error $e_8(x)$ using these M values. These estimates have then been used to construct the corrected solutions $y_{8,10}(x)$ and $y_{8,13}(x)$. In Table 2, the error estimations corresponding to $M = 10$ and $M = 13$ are given along with the actual errors of the corrected solutions $y_{8,10}(x)$ and $y_{8,13}(x)$ for several values of x. Table 2 makes it clear that the accuracy of the improved solutions is a direct consequence of the accuracy of error estimations, which is in parallel with the remark made at the end of Section 3.1. It is also seen that the

Table 1. Comparison of the absolute errors of the approximate solutions of Example 1 obtained by the present method, the Taylor method [24], Hermite method [23], Chebyshev method [25] and CMBOM method [14] for various values of N.

t	Taylor method with $N=5$	CMBOM method with $N=5$	Present method with $N=5$	Present method with $N=11$
0.2	0.854E − 7	0.369E − 6	0.368E − 5	0.103E − 13
0.4	0.536E − 5	0.237E − 5	0.121E − 4	0.617E − 13
0.6	0.595E − 4	0.596E − 5	0.120E − 4	0.140E − 12
0.8	0.326E − 3	0.348E − 4	0.397E − 5	0.260E − 12
1	0.121E − 2	0.203E − 3	0.575E − 6	0.411E − 12
t	Chebyshev method with $N=8$	CMBOM method with $N=8$	Hermite method with $N=8$	Present method with $N=8$
0.2	0.370E − 6	0.511E − 10	0.620E − 8	0.774E − 10
0.4	0.238E − 5	0.250E − 9	0.576E − 7	0.313E − 10
0.6	0.597E − 5	0.594E − 9	0.179E − 6	0.537E − 9
0.8	0.348E − 4	0.711E − 9	0.373E − 6	0.693E − 9
1	0.203E − 3	0.268E − 7	0.636E − 6	0.117E − 8

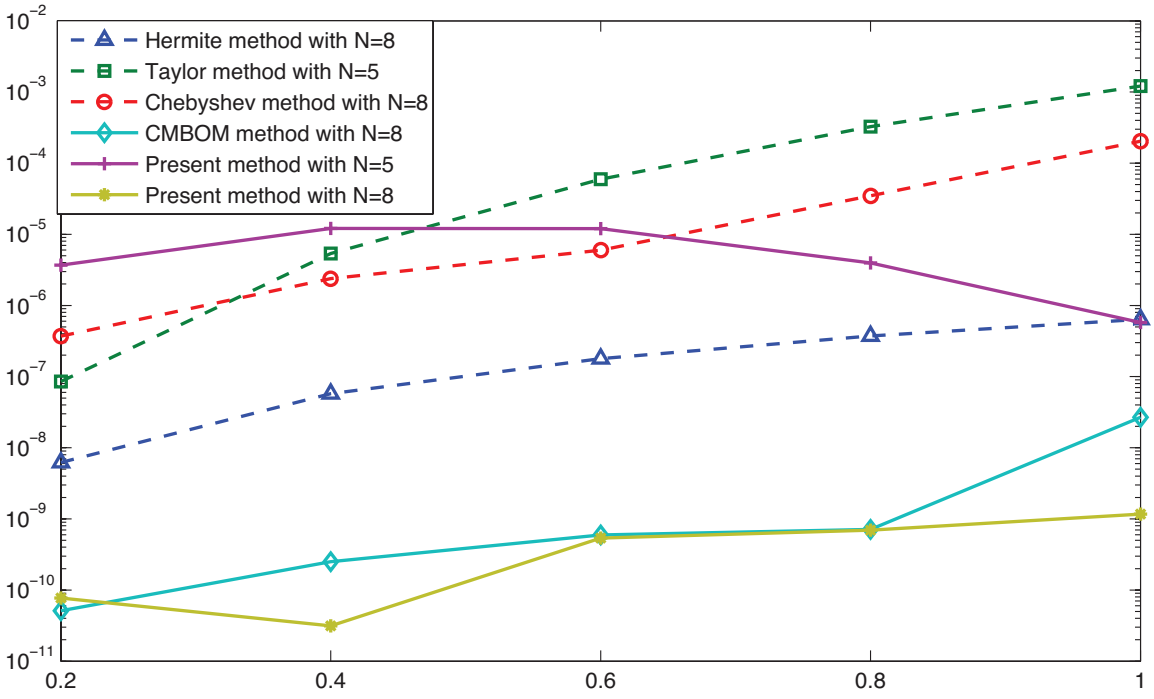

Figure 1. Graphics of the absolute errors of Hermite polynomial method, Taylor method, Chebyshev method, CMBOM method, and the present method for different values of N in Example 1.

estimates corresponding to $M=13$ are more accurate than those corresponding to $M=10$, which is because the proposed method is more accurate when implemented using higher-degree polynomials. These remarks can also be confirmed from Figure 2, where the data in Table 2 are depicted graphically.

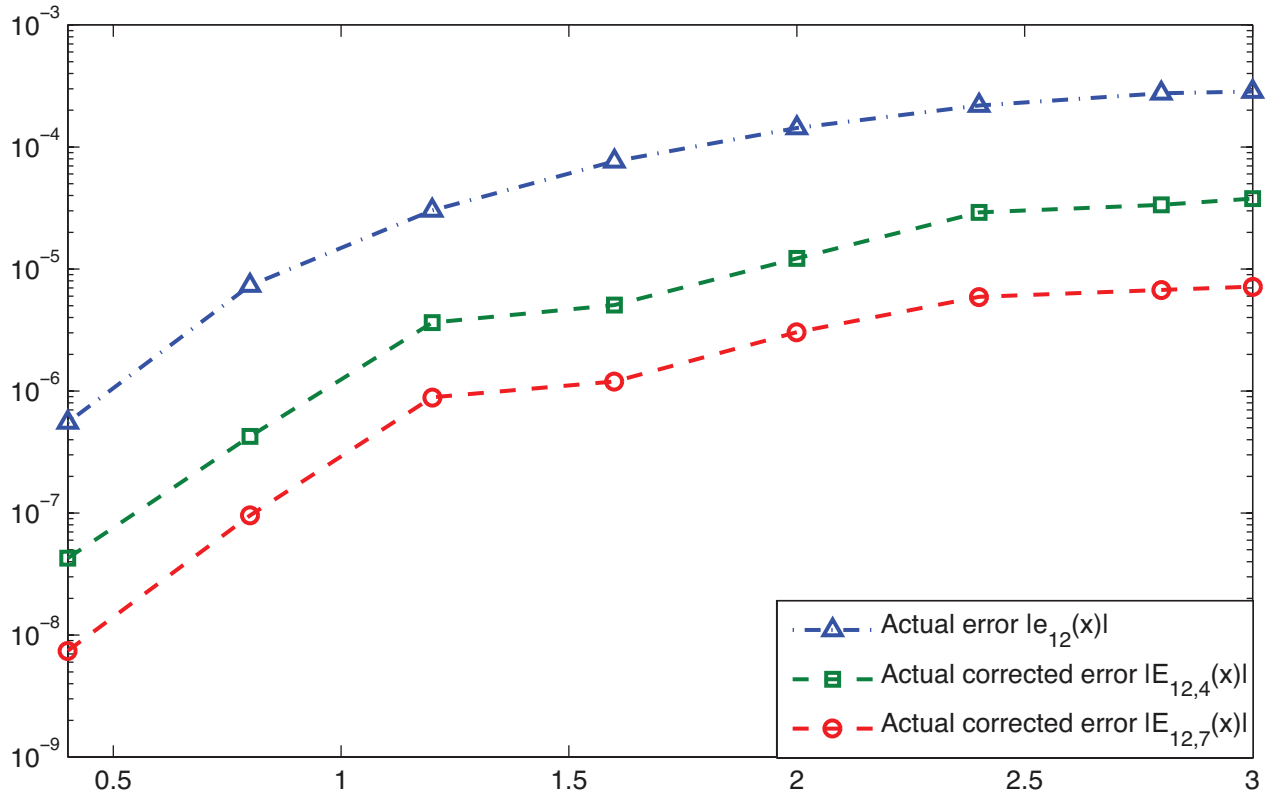

Figure 2. Comparison of the actual error $e_8(x)$ with the estimated errors $e_{8,10}(x)$ and $e_{8,13}(x)$ and with the corrected errors $E_{8,10}(x)$ and $E_{8,13}(x)$ of its two improvements in Example 1.

As an illustration of the meaning of the error bound (3.6) proved in Section 3.2, let us consider the approximate solutions of equation (4.1) obtained with $N = 5, N = 8$ and $N = 11$. The error bound (3.6) consists of two parts, which are the Taylor truncation error of the exact solution e^{-x} and the maximum distance of the approximate solution to the Taylor polynomial having the same degree. The N-th degree Taylor polynomial of the exact solution e^{-x} around $x = 0$ is given by

$$y_N^T(x) = 1 - x + \frac{x^2}{2} - \frac{x^3}{6} + \ldots + (-1)^N \frac{x^N}{N!}$$

and its absolute truncation error is given by

$$|R_N^T(x)| = \frac{x^{N+1}}{(N+1)!} \left| \frac{d^{N+1}(e^{-x})}{dx^{N+1}} \right|_{x=c_x} = \frac{x^{N+1}}{(N+1)!} e^{-c_x} < \frac{1}{(N+1)!}$$

since we have $0 < c_x < x < 1$ and $\dfrac{d^{N+1}}{dx^{N+1}} e^{-x} = (-1)^{N+1} e^{-x}$. As for the second part of the error bound (3.6), we first compute the maximum distance between the fifth degree Taylor polynomial

$$y_N^T(x) = 1 - x + \frac{x^2}{2} - \frac{x^3}{6} + \frac{x^4}{24} - \frac{x^5}{120}$$

of e^{-x} around $x = 0$ and the approximate solution

$$y_5(x) = 1 - x + 0.5x^2 - 0.1657425668x^3 + 0.0387656280x^4 - 0.0051430446x^5.$$

Table 2. Comparison of the actual absolute errors for $y_8(x)$ and its two improvements for $M = 10$ and $M = 13$ in Example 1.

x	Actual error for $y_8(x)$ $N = 8$	Estimated error for $y_{8,10}(x)$ $N = 8, M = 10$	Estimated error for $y_{8,13}(x)$ $N = 8, M = 13$	Actual error for $y_{8,10}(x)$ $N = 8, M = 10$	Actual error for $y_{8,13}(x)$ $N = 8, M = 13$
0	0	0	0	0	0
0.2	0.774E − 10	0.773E − 10	0.774E − 10	0.107E − 12	0
0.4	0.313E − 10	0.305E − 10	0.313E − 10	0.790E − 12	0
0.6	0.537E − 9	0.535E − 9	0.537E − 9	0.166E − 11	0.333E − 15
0.8	0.693E − 9	0.690E − 9	0.693E − 9	0.321E − 11	0.666E − 15
1	0.117E − 8	0.116E − 8	0.117E − 8	0.502E − 11	0.777E − 15

Table 3. Comparison of the actual maximum absolute errors and the error bounds computed using (3.6) corresponding to $N = 5, 8$ and 11 in Example 1.

Max. error for $N = 5$	Err. bound for $N = 5$	Max. error for $N = 8$	Err. bound for $N = 8$	Max. error for $N = 11$	Err. bound for $N = 11$
0.136×10^{-4}	0.0026	0.117×10^{-8}	0.525×10^{-5}	0.286×10^{-10}	0.402×10^{-8}

This maximum distance occurs at the point $x = 1$, which is equal to 0.0012. Since the absolute value $|R_N^T(x)|$ of the Taylor truncation error is known to be bounded by $\frac{1}{6!} < 0.0014$, the error bound formula for $N = 5$ gives $|y(x) - y_5(x)| < 0.0026$. This procedure has also been carried out for the values $N = 8$ and $N = 11$. The obtained bounds are shown in Table 3 together with the maximum actual errors $|e_N(x)|$ corresponding to these N values. It is understood that the error bounds for this problem are rather loose since the actual maximum error values are much smaller in reality. It should be stressed once more that the error bound (3.6) is not an a priori error bound; it only shows that the actual error is bounded in part by the Taylor truncation error of the actual solution.

Example 4.2 *Our second example is the following third order generalized pantograph equation from [13], whose exact solution is* $y(x) = e^{-x}$.

$$y^{(3)}(x) - xy''(\frac{x}{3} - 1) + xy'(\frac{x}{4} + 1) + y(x) = -x(e^{-\frac{x}{3}+1} + e^{-\frac{x}{4}-1}),\ 0 \le x \le 3 \tag{4.2}$$
$$y(0) = 1, y'(0) = -1, y''(0) = 1.$$

We applied residual correction to $y_{12}(x)$ for the values $M = 4$ and $M = 7$. Table 4 and Figure 3 illustrate the absolute error values of the original approximation $y_{12}(x)$ and of the improvements $y_{12,4}(x)$ and $y_{12,7}(x)$ for several values of x. It can be commented looking at the table that residual correction greatly reduces the absolute error also for this problem. When we examine the two graphs in the figure, just like as in the first example problem, we see that the estimated absolute errors $|e_{12,4}(x)|$ and $|e_{12,7}(x)|$ are very close to the actual error $|e_{12}(x)|$, which explains the significant decrease in the actual corrected errors $|E_{12,4}(x)|$ and $|E_{12,7}(x)|$.

Table 4. Comparison of the actual absolute errors for $y_{12}(x)$ and its two improvements for $M = 4$ and $M = 7$ in Example 2.

x	Actual error for $y_{12}(x)$ $N = 12$	Estimated error for $y_{12,4}(x)$ $N = 12, M = 4$	Estimated error for $y_{12,7}(x)$ $N = 12, M = 7$	Actual error for $y_{12,4}(x)$ $N = 12, M = 4$	Actual error for $y_{12,7}(x)$ $N = 12, M = 7$
0	0	0	0	0	0
0.4	0.554E − 6	0.596E − 6	0.561E − 6	0.427E − 7	0.745E − 8
0.8	0.733E − 5	0.775E − 5	0.742E − 5	0.425E − 6	0.957E − 7
1.2	0.302E − 4	0.339E − 4	0.311E − 4	0.364E − 5	0.888E − 6
1.6	0.763E − 4	0.813E − 4	0.775E − 4	0.506E − 5	0.120E − 5
2.0	0.143E − 3	0.156E − 3	0.147E − 3	0.122E − 4	0.304E − 5
2.4	0.219E − 3	0.248E − 3	0.225E − 3	0.291E − 4	0.592E − 5
2.8	0.276E − 3	0.309E − 3	0.283E − 3	0.336E − 4	0.675E − 5
3.0	0.285E − 3	0.323E − 3	0.292E − 3	0.377E − 4	0.718E − 5

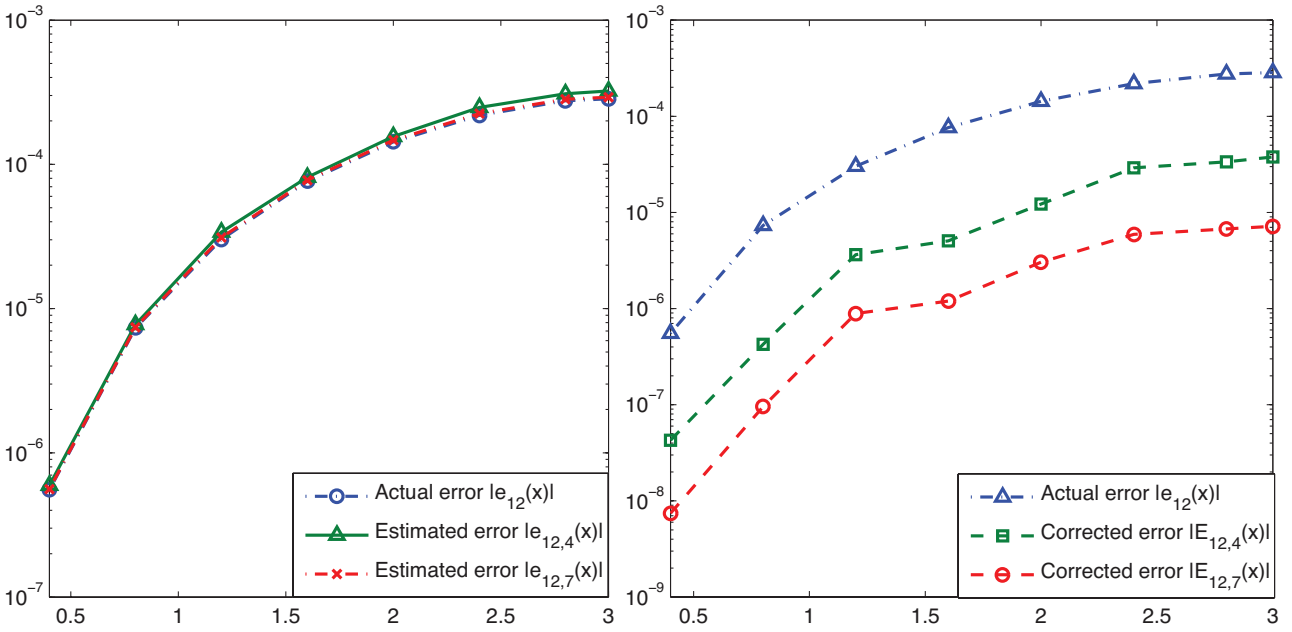

Figure 3. Comparison of the actual absolute error $|e_{12}(x)|$ with the estimated errors $|e_{12,4}(x)|$ and $|e_{12,7}(x)|$ and with the corrected errors $|E_{12,4}(x)|$ and $|E_{12,7}(x)|$ of its two improvements in Example 2.

Example 4.3 *Next, we consider the following second-order generalized pantograph equation studied in [14]:*

$$y''(x) = 0.75y(x) + y(0.5x) - x^2 + 2,\ 0 \le x \le 1 \tag{4.3}$$
$$y(0) = 0, y'(0) = 0.$$

Implementing the present method with $N = 2$ yields the solution $y_2(x) = x^2$, which is the exact solution. In fact, for any choice of $N > 2$ we get $a_k = 0$ for $2 < k \le N$ as a result of the algorithm, which means any $N \ge 2$ yields the exact solution $y(x) = x^2$. This is not a surprise since the scheme described in Section 2 makes it clear that the unknown coefficients of the approximate solution $y_N(x)$ obtained as a result are equal to the

actual coefficients of the exact solution. Therefore, this example application shows that the present method yields the exact solution in case this solution is a polynomial.

5. Conclusions

In this paper, we have presented a Galerkin-like approach for the approximate solution of linear functional differential equations. The method includes taking inner product of an expression derived from equation (1.2) with a set of monomials. The results of this approach are compared with some popular methods present in the literature and it is revealed that the present method performs slightly better than the others in terms of absolute error. Residual error correction technique to improve the accuracy of approximate solutions has also been discussed. Simulation results show that significant improvements in the approximate solutions can be achieved as a result of employing this technique. It is stressed that this fact can be attributed to the accuracy of our error estimation related to the approximate solution. Lastly, the presented method implemented with parameter N gives rise to the exact solution in case the exact solution is a polynomial of degree at most N.

Acknowledgments

The authors are supported by the Scientific Research Project Administration of Akdeniz University. In addition, the authors are grateful to two anonymous referees who helped to greatly improve the quality of this manuscript with their constructive comments.

References

[1] Ockendon JR, Tayler AB. The dynamics of a current collection system for an electric locomotive. Proceedings of the Royal Society of London A 1971; 322: 447-468. doi: 10.1098/rspa.1971.0078

[2] Tayler AB. Mathematical Models in Applied Mechanics. New York, NY, USA: Oxford University Press, 1985.

[3] Brunner H. Collocation Methods for Volterra Integral and Related Functional Equations. Cambridge, UK: Cambridge University Press, 2004.

[4] Derfel GA, Vogl F. On the asymptotics of solutions of a class of linear functional-differential equations. European Journal of Applied Mathematics 1996; 7 (5): 511-518. doi: 10.1017/S0956792500002527

[5] Feldstein A. On neutral functional-differential equations with variable time delays. Mathematical Proceedings of the Cambridge Philosophical Society 1998; 124 (2): 371-398. doi: 10.1017/S0305004198002497

[6] Dehghan M, Shakourifar M, Hamidi A. The solution of linear and non-linear systems of Volterra functional equations using Adomian-Padé technique. Chaos Solitons & Fractals 2009; 39 (5): 2509-2521. doi: 10.1016/j.chaos.2007.07.028

[7] Feng X. An analytic study on the multi-pantograph delay equations with variable coefficients. Bulletin Mathématiques de la Société des Sciences Mathématiques de Roumanie 2013; 56 (2): 205-215.

[8] Yüzbaşı Ş, Karaçayır M. A Galerkin-like approach to solve multi-pantograph type delay differential equations. FILOMAT 2018; 32 (2): 409-422. doi: 10.2298/FIL1802409Y

[9] Ishiwata E, Muroya Y. On collocation methods for delay differential and Volterra integral equations with proportional delay. Frontiers of Mathematics in China 2009; 4 (1): 89-111. doi: 10.1007/s11464-009-0004-x

[10] Işık OR, Güney Z, Sezer M. Bernstein series solutions of pantograph equations using polynomial interpolation. Journal of Difference Equations and Applications 2012; 18 (3): 357-374. doi: 10.1080/10236198.2010.496456

[11] Yüzbaşı Ş, Şahin N, Sezer M. A Bessel collocation method for numerical solution of generalized pantograph equations. Numerical Methods for Partial Differential Equations 2011; 28 (4): 1105-1123. doi: 10.1002/num.20660

[12] Sedaghat S, Ordokhani Y, Dehghan M. Numerical solution of the delay differential equations of pantograph type via Chebyshev polynomials. Communications in Nonlinear Science and Numerical Simulation 2012; 17 (12): 4815-4830. doi: 10.1016/j.cnsns.2012.05.009

[13] Yüzbaşı Ş, Sezer M. An exponential approximation for solutions of generalized pantograph-delay differential equations. Applied Mathematical Modelling 2013; 37 (22): 9160-9173. doi: 10.1016/j.apm.2013.04.028

[14] Tohidi E, Bhrawy AH, Erfani K. A collocation method based on Bernoulli operational matrix for numerical solution of generalized pantograph equation. Applied Mathematical Modelling 2013; 37 (6): 4283-4294. doi: 10.1016/j.apm.2012.09.032

[15] Sezer M, Yalçınbaş S, Gülsu M. A Taylor polynomial approach for solving generalized pantograph equations with nonhomogenous term. International Journal of Computer Mathematics 2011; 85 (7): 1055-1063. doi: 10.1080/00207160701466784

[16] Baykuş Savaşaneril N, Sezer M. Hybrid Taylor-Lucas collocation method for numerical solution of high-order pantograph type delay differential equations with variable delays. Applied Mathematics & Information Sciences 2017; 11 (6): 1795-1801. doi: 10.18576/amis/110627

[17] Saadatmandi A, Dehghan M. Variational iteration method for solving a generalized pantograph equation. Computers & Mathematics with Applications 2009; 58 (11-12): 2190-2196. doi: 10.1016/j.camwa.2009.03.017

[18] Yüzbaşı Ş, İsmailov N. A Taylor operation method for solutions of generalized pantograph type delay differential equations. Turkish Journal of Mathematics 2018; 42: 395-406. doi: 10.3906/mat-1506-71

[19] Javadi S, Babolian E, Taheri Z. Solving generalized pantograph equations by shifted orthonormal Bernstein polynomials. Journal of Computational and Applied Mathematics 2016; 303: 1-14. doi: 10.1016/j.cam.2016.02.025

[20] Gümgüm S, Baykuş Savaşaneril N, Kürkçü ÖK, Sezer M. Lucas polynomial solution of nonlinear differential equations with variable delays. Hacettepe Journal of Mathematics & Statistics 2019; doi:10.15672/hujms.460975

[21] Türkyılmazoğlu M. An effective approach for numerical solutions of high-order Fredholm integro-differential equations. Applied Mathematics and Computation 2014; 227: 384-398. doi: 10.1016/j.amc.2013.10.079

[22] Yüzbaşı Ş, Karaçayır M. A Galerkin-like approach to solve high-order integro-differential equations with weakly singular kernel. Kuwait Journal of Science 2016; 43 (2): 61-75.

[23] Sezer M, Akyüz Daşçıoğlu A. A Taylor method for numerical solution of generalized pantograph equations with linear functional argument. Journal of Computational and Applied Mathematics 2007; 200 (1): 217-225. doi: 10.1016/j.cam.2005.12.015

[24] Yalçınbaş S, Aynıgül M, Sezer M. A collocation method using Hermite polynomials for approximate solution of pantograph equations. Journal of the Franklin Institute 2011; 348 (6): 1128-1139. doi: 10.1016/j.jfranklin.2011.05.003

[25] Öztürk Y, Gülsu M. Approximate solution of linear generalized pantograph equations with variable coefficients on Chebyshev-Gauss grid. Journal of Advanced Research in Scientific Computing 2012; 4 (1): 36-51.