Turkish Journal of Mathematics

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

Research Article

Turk J Math
(2018) 42: 243 – 256
© TÜBİTAK
doi:10.3906/mat-1611-126

An operational matrix method for solving linear Fredholm–Volterra integro-differential equations

Şuayip YÜZBAŞI*, **Nurbol ISMAILOV**
Department of Mathematics, Faculty of Science, Akdeniz University, Antalya, Turkey

Received: 30.11.2016 • **Accepted/Published Online:** 21.04.2017 • **Final Version:** 22.01.2018

Abstract: The aim of this paper is to propose an efficient method to compute approximate solutions of linear Fredholm–Volterra integro-differential equations (FVIDEs) using Taylor polynomials. More precisely, we present a method based on operational matrices of Taylor polynomials in order to solve linear FVIDEs. By using the operational matrices of integration and product for the Taylor polynomials, the problem for linear FVIDEs is transformed into a system of linear algebraic equations. The solution of the problem is obtained from this linear system after the incorporation of initial conditions. Numerical examples are presented to show the applicability and the efficiency of the method. Wherever possible, the results of our method are compared with those yielded by some other methods.

Key words: Integro-differential equations, operational matrix method, Taylor polynomials, inner product, best polynomial approximation

1. Introduction

Various problems of physics and engineering lead to investigation of integro-differential equations. For instance, Tadmor and Athavale [25] introduced a class of integro-differential equations to model images. In addition, a particularly rich source of such equations is electrical circuit analysis [2]. For this reason, the theory of these equations has attracted the attention of physicists and mathematicians for a long time. Many papers are devoted to the problems of the existence, uniqueness and stability of solutions; the regularization of ill-posed problems; and the analytical and numerical solution methods of the above mentioned equations [7, 16, 26].

Since obtaining the exact solution of these equations is difficult, researchers have so far developed many numerical methods. Some of these methods are the homotopy approach [14], the Chebyshev polynomial approach [1], the collocation method [34], the Legendre wavelets method [20], the Taylor polynomials method [31], the Tau method [11], the compact finite difference method [35], the multi-parametric homotopy approach [15], a topological method [23], a method based on the power series and Pade series [27], a Taylor collocation method [4], the Jacobi-spectral method [3], and the methods in [18–21].

In [6, 10, 19, 32], operational matrix methods based on Bernstein, Jacobi, and Chebyshev polynomials were presented in order to numerically solve integro-differential equations. Our main purpose is to apply operational matrices of integration and product for VFIDEs. For that purpose, we utilize the least-squares approximation of the given function by Taylor polynomials to obtain operational matrices. Unknown function and its derivatives are expressed in matrix forms by using operational matrices. The resulting matrix forms are

*Correspondence: syuzbasi@akdeniz.edu.tr

substituted in the given functional equation. Then we drop out the matrix depending on the unknown variable. The remaining expression is the linear system of algebraic equations to be solved. Finally, solving this system, we get the coefficients of the polynomial solution belonging to the considered functional equation.

We consider the mth-order linear FVIDE given by

$$\sum_{k=0}^{m}\varphi_k(x)\,u^{(k)}(x)=g(x)+\lambda_1\int_0^1 K_1(x,t)\,u(t)\,dt+\lambda_2\int_0^x K_2(x,t)\,u(t)\,dt\ \ (0\leq x,t\leq 1)\,, \tag{1}$$

with initial conditions

$$u^{(i)}(0)=\alpha_i\ \ (i=0,1,...,m-1)\,.$$

Here $\varphi_k(x)$, $g(x)$, $K_1(x,t)$, and $K_2(x,t)$ are smooth functions on the interval $0\leq x,t\leq 1$, α_i, λ_1 and λ_2 are given real constants. Let us define

$$I_1=\sum_{k=0}^{m}\varphi_k(x)\,u^{(k)}(x)\,,I_2=\int_0^1 K_1(x,t)\,u(t)\,dt, I_3=\int_0^x K_2(x,t)\,u(t)\,dt.$$

Our aim is to approximate each part I_1, I_2, I_3 of equation (1) by truncated Taylor series. To this end, we outline operational matrices for Taylor polynomials.

This paper is organized as follows: in the next section, we introduce the operational matrices. The solution method is given in Section 3. Section 4 is devoted to the demonstration of the validity of the present method. Finally, Section 5 contains conclusions regarding the present scheme.

2. Operational matrices

In this section, we first give preliminary theorems for least-squares approximation of function by polynomials. Then we use it to derive operational matrices of integration and product. The theorems that we will mention are also stated and proved in [17]. In the rest of this paper, we will use the term "Taylor polynomials" in order to refer to polynomials expressed in the standard basis $\{1,x,x^2,\ldots,x^N\}$ for some natural number N. We emphasize that the results that will be obtained by the present method are not the same as those obtained by the Taylor polynomials method.

2.1. Least-squares approximation

Let V be a normed space and suppose that any given element $v\in V$ is to be approximated by an element $w\in W$, where W is a fixed subspace of V. Moreover, M is a complete subspace of W.

Theorem 1 *Let V be an inner product space and $M\neq\emptyset$ a convex subset that is complete (in the metric induced by the inner product). Then for every given $v\in V$ there exists a unique $w\in M$ such that*

$$\delta=\inf_{w^*\in M}\|v-w^*\|=\|v-w\|\,.$$

Lemma 1 *In Theorem 1, let M be a complete subspace of W and $v\in V$ be fixed. Then $z=v-w$ is orthogonal to W.*

Now let $V = L^2\left[0,1\right]$. Let us define the inner product by

$$\langle f, g\rangle = \int_0^1 f(x)g(x)dx$$

and suppose that any given $f \in L^2\left[0,1\right]$ is to be approximated by $w \in W$, where $W = Span\{1, x, x^2, ..., x^N\}$. Any element $w_0 \in W$ is uniquely determined by the coefficients $\mathbf{A} = [a_0, a_1, ..., a_N]$ such that

$$f(x) \approx w_0 = \sum_{i=0}^{N} a_i x^i = \mathbf{A}\mathbf{X}\left(x\right), \tag{2}$$

where $\mathbf{X}\left(x\right) = [1\ x\ x^2 ...\ x^N]^T$. Therefore, W is finite dimensional and it is a complete subspace too. In view of Theorem 1, there is a best approximation out of W. Let w_0 be the best approximation of f, that is

$$\|f - w_0\|_2 \leq \|f - w\|_2$$

for all $w \in W$, where $\|f\|_2 = \left(\int_0^1 f^2(x)dx\right)^{1/2}$. According to Lemma 2, $f - w_0$ is orthogonal to W. This implies that

$$\int_0^1 (f(x) - w_0)x^i dx = 0$$

for $i = 0, 1, ..., N$. The last equations can be written as

$$\mathbf{A}\int_{\mathbf{0}}^{\mathbf{1}} \mathbf{X}(x)\mathbf{X}^T(x)dx = \int_0^1 f(x)\mathbf{X}^T(x)dx.$$

Then $\mathbf{A}$ can be obtained by

$$\mathbf{A} = \int_0^1 f(x)\mathbf{X}^T(x)dx \left(\int_0^1 \mathbf{X}(x)\mathbf{X}^T(x)dx\right)^{-1},$$

where

$$\int_0^1 f(x)X^T(x)dx = \left[\int_0^1 f(x)dx,\ \int_0^1 xf(x)dx, ...,\ \int_0^1 x^N f(x)dx\right].$$

Let

$$\mathbf{Q} = \int_0^1 \mathbf{X}(x)\mathbf{X}^T(x)dx.$$

Then, in view of the integrals in the last expression, we find

$$\begin{aligned}\mathbf{Q} &= [Q_{i,j}] = \left[\int_0^1 x^{i-1}x^{j-1}dx\right] \\ &= \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{N+1} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{N+2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{1}{N+1} & \frac{1}{N+2} & \frac{1}{N+3} & \cdots & \frac{1}{2N+1} \end{bmatrix}\end{aligned}$$

where $i,j = 1,...,N+1$, which is the well-known Hilbert matrix. Although the condition number of the Hilbert matrix is very high, this situation does not have a negative effect on the solution of the above linear system since we actually need the inverse of $\mathbf{Q}$ rather than $\mathbf{Q}$ itself. The exact inverse of the Hilbert matrix of any size can be readily obtained using any computer algebra system. For instance, one can make use of the Symbolic Math Toolbox in MATLAB to compute it without error by first defining the Hilbert matrix of size N by `H=sym(hilb(N))` and then finding its inverse by `inv(H)`. For more detail, see the note in [30, Ex. 4.22].

2.2. Operational matrix of integration for Taylor polynomials

To establish the relation between the unknown function and its derivatives, we use the operational matrix of integration. In this subsection, we obtain the operational matrix of integration for Taylor polynomials.

Suppose that P is an $(N+1)\times(N+1)$ operational matrix of integration; then

$$\int_0^x \mathbf{X}(t)\,dt \approx \mathbf{P}\mathbf{X}(x),$$

where $0 \le x \le 1$.

Integrating the components of $\mathbf{X}(t)$, and defining two new matrices as

$$\mathbf{S} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{2} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & \frac{1}{N+1} \end{bmatrix}, \ \mathbf{X}_1(x) = \begin{bmatrix} x \\ x^2 \\ \vdots \\ x^{N+1} \end{bmatrix},$$

we have the following product of matrices:

$$\int_0^x \mathbf{X}(t)\,dt = \mathbf{S}\mathbf{X}_1(x) = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{2} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & \frac{1}{N+1} \end{bmatrix} \begin{bmatrix} x \\ x^2 \\ \vdots \\ x^{N+1} \end{bmatrix}.$$

Components of $\mathbf{X}_1(x)$ can be expressed in terms of the basis set $\mathbf{X}(x)$. We can write

$$\mathbf{X}(x) = \mathbf{I}\mathbf{X}(x),$$

where $\mathbf{I}$ is the $(N+1)\times(N+1)$ identity matrix. Then

$$x^{i-1}=\mathbf{I}_i\mathbf{X}(x)\ (i=1,2,...,N+1),$$

where $\mathbf{I}_i$ is the $i-$th row of $\mathbf{I}$. For $f(x)=x^{N+1}$, by (2) we have

$$x^{N+1}\approx\mathbf{A}_{N+1}\mathbf{X}(x).$$

Now, substituting the expressions for $x^i\ \ i=1,2,...,N+1$, in $\mathbf{X}_1(x)$ we have

$$\int_0^x\mathbf{X}(t)\,dt\approx\mathbf{SLX}(x).$$

Here $\mathbf{L}=\left[\mathbf{I}_2^T\ \mathbf{I}_3^T\ ...\ \mathbf{I}_{N+1}^T\ \mathbf{A}_{N+1}^T\right]^T$ is an $(N+1)\times(N+1)$ matrix. Finally, we get the operational matrix of integration in the following form:

$$\mathbf{P}=\mathbf{SL}. \tag{3}$$

2.3. Operational matrix of product for Taylor polynomials

It is frequently necessary to evaluate the product of vector $\mathbf{X}(x)$ and the given function. In this subsection, we get the operational matrix of product, which is very useful when considering a functional equation that has variable coefficients. Let $\alpha(x)\in L^2[0,1]$ be a given function; then $\mathbf{C}$ is an $(N+1)\times(N+1)$ operational matrix of product and the relation

$$\alpha(x)\mathbf{X}(x)\approx\mathbf{CX}(x)$$

holds.

Now we want to approximate elements of the product $\alpha(x)\mathbf{X}(x)$ using the scheme that was described at the beginning of this section.

By (2), we have

$$x^{i-1}\alpha(x)\approx\mathbf{E}_i\mathbf{X}(x).\ \ (i=1,2,...,N+1)$$

Then

$$\alpha(x)\mathbf{X}(x)=\begin{bmatrix}\alpha(x)\\ x\alpha(x)\\ \vdots\\ x^N\alpha(x)\end{bmatrix}\approx\begin{bmatrix}\mathbf{E}_1\\ \mathbf{E}_2\\ \vdots\\ \mathbf{E}_{N+1}\end{bmatrix}\mathbf{X}(x).$$

Therefore, our desired matrix is

$$\mathbf{C}=\begin{bmatrix}\mathbf{E}_1\\ \mathbf{E}_2\\ \vdots\\ \mathbf{E}_{N+1}\end{bmatrix}. \tag{4}$$

In the next section, we utilize operational matrices to solve FVIDEs.

3. Description of the method

In many matrix methods [1, 31, 34], firstly the unknown function is expressed by polynomials, then its derivatives are determined by using the operational matrix of derivatives. In our case, we do the opposite, i.e. we firstly express the high derivative of the unknown function by Taylor polynomials. Then, by using the operational matrix of integration and the initial conditions, we express the unknown function by polynomials. Thus, let

$$u^{(m)}(x) \approx u_N^{(m)}(x) = \mathbf{A}\mathbf{X}(x). \tag{5}$$

Then, integrating (5) by using matrix integration (3) and the Newton–Leibniz formula, we have

$$\begin{aligned}
u_N^{(m-1)}(x) - u_N^{(m-1)}(0) &= u_N^{(m-1)}(x) - \alpha_{m-1} = \mathbf{A}\int\limits_{\mathbf{0}}^{\mathbf{x}} \mathbf{X}(t)\,dt = \mathbf{A}\mathbf{P}\mathbf{X}(x), \qquad (6)\\
u_N^{(m-2)}(x) &= \mathbf{A}\mathbf{P}^2\mathbf{X}(x) + \alpha_{m-1}x + \alpha_{m-2},\\
&\ldots\\
u_N^{(m-l)}(x) &= \mathbf{A}\mathbf{P}^l\mathbf{X}(x) + \sum_{i=0}^{l-1}\frac{\alpha_{m-l+i}x^i}{i!},\\
&\ldots\\
u_N(x) &= \mathbf{A}\mathbf{P}^m\mathbf{X}(x) + \sum_{i=0}^{m-1}\frac{\alpha_i x^i}{i!} \ \ (l = 1, ..., m).
\end{aligned}$$

Substituting the expressions (5) and (6) in I_1, we have

$$\begin{aligned}
I_1 &= \sum_{k=0}^{m}\varphi_k(x)\,u^{(k)}(x) = \sum_{k=0}^{m-1}\varphi_k(x)\left[\mathbf{A}\mathbf{P}^{m-k}\mathbf{X}(x) + \sum_{i=0}^{m-k-1}\frac{\alpha_{k+i}x^i}{i!}\right] + \varphi_m(x)\,\mathbf{A}\mathbf{X}(x)\\
&= \sum_{k=0}^{m-1}\mathbf{A}\mathbf{P}^{m-k}\left[\varphi_k(x)\,\mathbf{X}(x)\right] + \mathbf{A}\left[\varphi_m(x)\,\mathbf{X}(x)\right] + \sum_{k=0}^{m-1}\sum_{i=0}^{m-k-1}\varphi_k(x)\,\frac{\alpha_{k+i}x^i}{i!}.
\end{aligned}$$

Then, using the operational matrix of product (4), the last equation can be expressed as

$$I_1 = \sum_{k=0}^{m}\mathbf{A}\mathbf{P}^{m-k}\mathbf{C}_k\mathbf{X}(x) + f_1(x), \tag{7}$$

where $f_1(x) = \sum\limits_{k=0}^{m-1}\sum\limits_{i=0}^{m-k-1}\varphi_k(x)\,\frac{\alpha_{k+i}x^i}{i!}$ and $\mathbf{C}_k$ are the operational matrices of product. For the second part, we have

$$\begin{aligned}
I_2 &= \int\limits_0^1 K_1(x,t)\,u(t)\,dt = \int\limits_0^1 K_1(x,t)\left[\mathbf{A}\mathbf{P}^m\mathbf{X}(t) + \sum_{i=0}^{m-1}\frac{\alpha_i t^i}{i!}\right]dt\\
&= \mathbf{A}\mathbf{P}^m\left[\int\limits_0^1 K_1(x,t)\,\mathbf{X}(t)\,dt\right] + \int\limits_0^1\sum_{i=0}^{m-1}K_1(x,t)\,\frac{\alpha_i t^i}{i!}dt.
\end{aligned}$$

Let $\mathbf{L}_1$ be the $(N+1)\times 1$ matrix defined by

$$\mathbf{L}_1 = \int_0^1 K_1(x,t)\,\mathbf{X}(t)\,dt.$$

We approximate each element of $\mathbf{L}_1$ by (2) as

$$\int_0^1 t^i K_1(x,t)\,dt \approx \mathbf{D}_i\mathbf{X}(x) \;\; (i=0,1,...,N),$$

and $\mathbf{L}_1$ can be expressed as follows:

$$\mathbf{L}_1 \approx \left[\mathbf{D}_0^T\ \mathbf{D}_1^T\ ...\ \mathbf{D}_N^T\right]^T \mathbf{X}(x) = \mathbf{DX}(x). \tag{8}$$

Now if we define

$$f_2(x) = \int_0^1 \sum_{i=0}^{m-1} K_1(x,t)\,\frac{\alpha_i t^i}{i!}dt$$

and use (8), we get

$$I_2 = \mathbf{AP}^m\mathbf{DX}(x) + f_2(x). \tag{9}$$

For the last part, we have

$$\begin{aligned} I_3 &= \int_0^x K_2(x,t)\,u(t)\,dt = \int_0^x K_2(x,t)\left[\mathbf{AP}^m\mathbf{X}(t) + \sum_{i=0}^{m-1}\frac{\alpha_i t^i}{i!}\right]dt \\ &= \mathbf{AP}^m\left[\int_0^x K_2(x,t)\,\mathbf{X}(t)\right] + \int_0^x\left(\sum_{i=0}^{m-1} K_2(x,t)\,\frac{\alpha_i t^i}{i!}\right)dt. \end{aligned}$$

Analogously, let $\mathbf{L}_2$ be the $(N+1)\times 1$ matrix defined by

$$\mathbf{L}_2 = \int_0^x K_2(x,t)\,\mathbf{X}(t)\,dt.$$

Then, by using (2) for each element of $\mathbf{L}_2$, we have

$$\int_0^x t^i K_2(x,t)\,dt \approx \mathbf{M}_i\mathbf{X}(x). \;\; (i=0,1,...,N)$$

Then we obtain

$$\mathbf{L}_2 = \left[\mathbf{M}_0^T\ \mathbf{M}_1^T\ ...\ \mathbf{M}_N^T\right]^T \mathbf{X}(x) = \mathbf{MX}(x). \tag{10}$$

Using equation (10), for I_3 we have

$$I_3 = \mathbf{A}\mathbf{P}^m\mathbf{M}\mathbf{X}(x) + f_3(x), \tag{11}$$

where $f_3(x) = \int_0^x \left(\sum_{i=0}^{m-1} K_2(x,t) \frac{\alpha_i t^i}{i!} \right) dt.$

Now, using (7), (9), and (11) we can write the approximation of equation (1), which is obtained with the help of operational matrices, in the following form:

$$\sum_{k=0}^{m} \mathbf{A}\mathbf{P}^{m-k}\mathbf{C}_k\mathbf{X}(x) = \lambda_1\mathbf{A}\mathbf{P}^m\mathbf{D}\mathbf{X}(x) + \lambda_2\mathbf{A}\mathbf{P}^m\mathbf{M}\mathbf{X}(x) + \mathbf{F}\mathbf{X}(x). \tag{12}$$

Here for $f(x) = g(x) + f_1(x) + f_2(x) + f_3(x)$ we can write

$$f(x) \approx \mathbf{F}\mathbf{X}(x).$$

To obtain the unknown coefficients, we drop out $\mathbf{X}(x)$ in equation (12). Provided that the matrix $\left(\sum_{k=0}^{m} \mathbf{P}^{m-k}\mathbf{C}_k - \lambda_1\mathbf{P}^m\mathbf{D} - \lambda_2\mathbf{P}^m\mathbf{M} \right)$ has full rank, i.e. it has rank $N+1$, then the unknown coefficients are determined by

$$\mathbf{A} = \mathbf{F}\mathbf{W}^{-1}, \tag{13}$$

where

$$\mathbf{W} = \sum_{k=0}^{m} \mathbf{P}^{m-k}\mathbf{C}_k - \lambda_1\mathbf{P}^m\mathbf{D} - \lambda_2\mathbf{P}^m\mathbf{M} \tag{14}$$

is the system matrix. Finally, using (6), we get the approximate solution $u_N(x)$ of equation (1).

4. Numerical examples

In this section, we apply the present method for some FVIDEs and compare the error of approximate solutions with other methods. All the numerical computations have been carried out using a program written in MATLAB.

Example 1 *[34] Our first problem is*

$$u''(x) + xu'(x) - xu(x) = \exp(x) - \sin x + 0.5x\cos x + \int_0^1 \sin x \exp(-t)\, u(t)\, dt - 0.5\int_0^x \cos x \exp(-t)\, u(t)\, dt, \tag{15}$$

with initial conditions $u(0) = u'(0) = 1.$

The exact solution of this problem is $u(x) = \exp(x)$. We wish to find the approximate solution of (15) for $N = 4$ by using the present method. To this end, we approximate the derivative as

$$u''(x) \approx u_4''(x) = \mathbf{A}\mathbf{X},$$

where $\mathbf{A} = [a_0, a_1, a_2, a_3, a_4]$, $\mathbf{X} = \begin{bmatrix} 1 & x & x^2 & x^3 & x^4 \end{bmatrix}$. Using (6), we have

$$\begin{aligned} u'(x) &\approx u_4'(x) = \mathbf{A}\mathbf{P}\mathbf{X} + 1 \\ u(x) &\approx u_4(x) = \mathbf{A}\mathbf{P}^2\mathbf{X} + x + 1, \end{aligned} \tag{16}$$

where

$$P=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0.333 & 0 \\ 0 & 0 & 0 & 0 & 0.25 \\ 7.936E-4 & -2.3809E-2 & 0.166 & -0.444 & 0.499 \end{bmatrix}$$

is the integration matrix. Moreover, the integral part is approximated as

$$\int_0^1 \sin x \exp\left(-t\right) u\left(t\right) dt \approx \mathbf{A}\mathbf{P}^2\mathbf{L}_1 + c_1 \sin x,$$

where $\mathbf{L}_1=\left[\sin x\int_0^1 \exp\left(-t\right) dt,\ \sin x\int_0^1 t\exp\left(-t\right) dt,\ ...,\ \sin x\int_0^1 t^N\exp\left(-t\right) dt\right]^T$, $c_1=\int_0^1 \left(t+1\right)\exp\left(-t\right) dt$. Using equation (2), we have

$$\mathbf{L}_1 \approx \mathbf{D}\mathbf{X}\left(x\right). \tag{17}$$

The integral with variable upper limit can be expressed as

$$\int_0^x \cos x \exp\left(-t\right) u\left(t\right) dt \approx \mathbf{A}\mathbf{P}^2\mathbf{L}_2 + \int_0^x \cos x\left(t+1\right)\exp\left(-t\right) dt,$$

where $\mathbf{L}_2=\left[\int_0^x \cos x \exp\left(-t\right) dt,\ \int_0^x t\cos x\exp\left(-t\right) dt, ..., \int_0^x t^N \cos x\exp\left(-t\right) dt\right]^T$. By using (10), the last matrix can be expressed as

$$\mathbf{L}_2 \approx \left[\mathbf{M}_0^T\ \mathbf{M}_1^T\ ...\ \mathbf{M}_N^T\right]^T\mathbf{X}\left(x\right)=\mathbf{M}\mathbf{X}\left(x\right). \tag{18}$$

Now, according to Section 2, the function

$$\begin{aligned} f\left(x\right) &= g\left(x\right)+f_1(x)+f_2(x)+f_3(x) \\ &= \exp\left(x\right)-\sin x+0.5x\cos x+x^2+c_1\sin x-0.5\int_0^x \cos x\left(t+1\right)\exp\left(-t\right) dt, \end{aligned}$$

where $f_1(x)=x^2$, $f_2\left(x\right)=c_1\sin x$, $f_3(x)=\int_0^x \cos x\left(t+1\right)\exp\left(-t\right) dt$ and $g\left(x\right)=\exp\left(x\right)-\sin x+0.5x\cos x$, is approximated as

$$f\left(x\right)\approx \mathbf{F}\mathbf{X}\left(x\right). \tag{19}$$

Substituting equations (16)–(19) in (14) and dropping out the matrix $\mathbf{X}\left(x\right)$, we get

$$\mathbf{A}=\mathbf{F}_1\left(\mathbf{I}+\mathbf{P}\mathbf{C}_1-\mathbf{P}^2\mathbf{C}_1-\mathbf{P}^2\mathbf{D}+0.5\mathbf{P}^2\mathbf{M}\right)^{-1},$$

where $\mathbf{C}_1$ is the matrix of product for $\alpha\left(x\right)=x$. Finally, using (16), we have the approximate solution

$$u_4\left(x\right)=0.069656x^4+0.139309x^3+0.510808x^2+0.998396x+1.00005.$$

The numerical results for the error function $e_N(x)$ and the approximate solution $u_N(x)$ are shown for $N = 9$ and $N = 10$ in Table 1. Since the approximate solution depends directly on the system matrix $\mathbf{W}$ in the system (13), the condition of this matrix is crucial in obtaining the coefficients of the final approximate solution correctly. For instance, the system matrix $\mathbf{W}$

$$\begin{bmatrix} 1.00007 & -0.08241 & 1.01372 & -0.43436 & -0.04297 \\ -0.00003 & 0.98210 & -0.00834 & 0.52832 & -0.17954 \\ -0.00035 & 0.00322 & 0.92661 & 0.19461 & 0.12206 \\ 0.00049 & -0.01849 & 0.10923 & 0.68242 & 0.42401 \\ 0.00143 & -0.04470 & 0.28882 & -0.70833 & 1.62648 \end{bmatrix}$$

for $N = 4$ has condition number 4.2272, while it has condition number 28.9494 and 55.6233 for $N = 9$ and $N = 10$, respectively.

Table 1. Exact and approximate values, Example 1.

x	Exact solution	Values of $u_N(x)$ N=9	Values of error $e_N(x)$ N=9	Values of error $e_N(x)$ N=10
0.2	1.221402758160170	1.221402758160681	5.111E-13	7.704E-13
0.4	1.491824697641270	1.491824697641063	2.077E-13	4.112E-13
0.6	1.822118800390509	1.822118800390078	4.310E-13	8.659E-14
0.8	2.225540928492468	2.225540928493186	7.175E-13	5.568E-13
1.0	2.718281828459046	2.718281828456610	2.435E-12	2.520E-13

Example 2 *[33] The second problem is the Fredholm integro-differential equation*

$$u'(x) = x\exp(x) + \exp(x) - x + \int_0^1 xu(t)\,dt$$

with the initial condition $u(0) = 0$.

The exact solution of this problem is $u(x) = x\exp(x)$. Applying the proposed method for $N = 5$, we get the approximate solution $u_5(x) = 0.0767445x^5 + 0.1223801x^4 + 0.5249685x^3 + 0.9935215x^2 + 1.0006671x - 0.0000163$. We also obtained the approximate solutions with $N = 3, 7, 10$. In Table 2, absolute errors of the present method corresponding to $N = 3, 5, 7$ are compared with errors of HPM [33] with $N = 4$ terms, Bessel collocation method [34] with $N = 5, 7$, CAS wavelet method [8] with $k = 2, M = 1$, and differential transform method [9] with $h = 0.1, n = 10$. Errors obtained by our method are smaller than those of the Bessel collocation method [34] and HPM [33] for listed parameter values, which is also shown visually in Figure 1. As for the CAS wavelet method [8] and differential transform method [9], although the errors of the present method are smaller, the related papers do not include any parameter values in order to present a fair comparison. In addition, Table 3 compares the errors of our solutions with $N = 10$ with those obtained by Bernoulli polynomials [5] with the same N value. The values demonstrate that errors of the present method are smaller for this N value. As for the condition of the system matrix, its condition number is equal to 1.7841, 1.7871, 1.7880, and 1.7884 for $N = 3, 5, 7$, and 10, respectively. These small numbers make us expect that little accuracy is lost when computing the coefficients of the approximate solutions for this example problem.

Table 2. Comparison of absolute errors for Example 2.

x	HPM [33] $N=4$	Bessel collocation [34]		Present method	
		N=5	N=7	N=5	N=7
0.2	0.925E-5	1.128E-5	2.266E-8	2.465E-6	6.105E-9
0.4	0.370E-4	7.916E-6	2.519E-8	8.795E-7	1.556E-9
0.6	0.833E-4	1.477E-5	2.732E-8	1.746E-6	5.914E-10
0.8	0.148E-3	5.405E-6	2.723E-8	3.151E-6	5.444E-9
x	CAS wavelet method [8]	Differential transformation [9]		Present method	
	$k=2, M=1$	$h=0.1, n=10$		$N=3$	
0.1	1.34917637E-3	1.00118319E-2		9.06437906E-4	
0.2	1.15960044E-3	2.78651355E-2		1.89226311E-3	
0.4	5.93105645E-2	7.55356316E-2		8.61682056E-4	
0.6	4.39287720E-2	1.09551714E-1		1.21866327E-3	
0.8	1.34514117E-2	6.94512700E-2		1.76527755E-3	
0.9	1.32045209E-2	1.00034260E-2		1.19827383E-3	

Table 3. Comparison of errors of Bernoulli polynomials [5] and present method, Example 2.

x	Bernoulli polynomials [5]		Present method	
	$u_{10}(x)$	$\|e_{10}(x)\|$	$u_{10}(x)$	$\|e_{10}(x)\|$
0.1	0.110517091657253	1.50312E-10	0.1105170918040005	3.56431E-12
0.2	0.244280551030778	6.01255E-10	0.2442805516251053	6.928703E-12
0.3	0.404957640919486	1.35331E-9	0.4049576422898417	1.70407E-11
0.4	0.596729876639513	2.41699E-9	0.5967298790466395	9.86865E-12
0.5	0.824360631451310	3.89875E-9	0.8243606353411325	8.93166E-12
0.6	1.093271273765889	6.46841E-9	1.093271280250227	1.59222E-11
0.7	1.409626882046905	1.31824e-8	1.409626895224665	4.6686E-12
0.8	1.780432707658582	3.51353e-8	1.780432742784639	9.33528E-12
0.9	2.213642693742780	1.06298E-7	2.213642800056560	1.53051E-11
1.0	2.718281510541929	3.17917E-7	2.718281828518279	5.92330E-11

Example 3 *[28] The next equation is*

$$u'(x) = 2\exp(x) - 2 + \int_0^x u(t)\,dt + \int_0^1 u(t)\,dt$$

with the initial condition $u(0) = 0$.

The exact solution of this problem is $u(x) = x\exp(x)$. The problem was solved by using the series solution method in [28]. We apply the present method for $N = 5, 8, 9$. The obtained numerical results are shown in Figure 2. The condition number of the system matrix is respectively 3.0609, 3.4784, 3.7972 for $N = 5, 8, 9$.

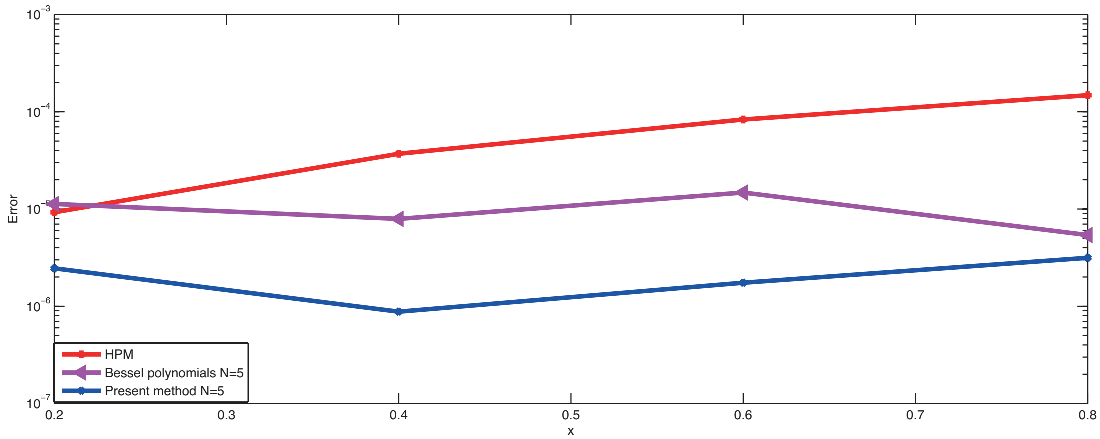

Figure 1. Absolute errors of the present method, homotopy perturbation method, and Bessel collocation method in Example 2.

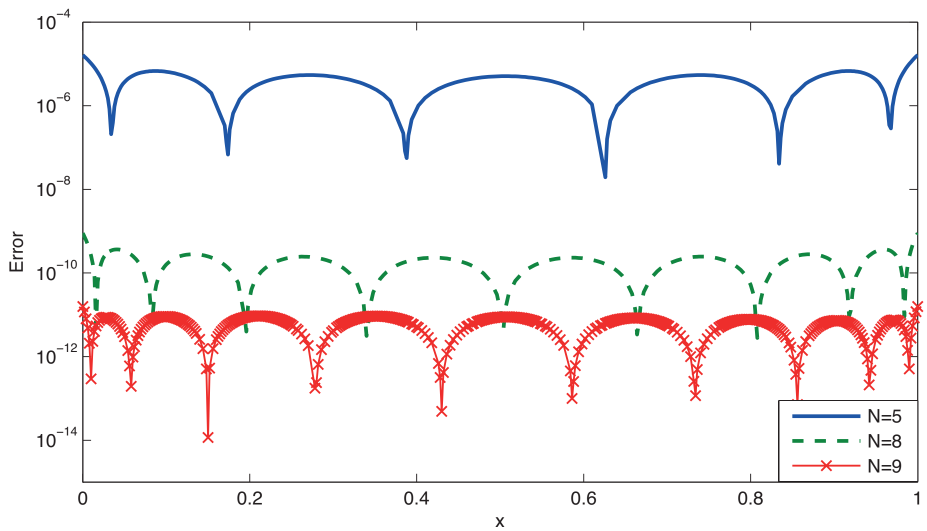

Figure 2. Absolute errors $|e_N(x)|$ for $N = 5, 8, 9$ in Example 3.

Table 4. Error values for Example 4.

x	Tau method [22]	Present method		
	N=10	N=10	N=12	N=14
0.2	1.00E-10	7.389E-15	9.473E-16	5.595E-1
0.4	1.00E-10	5.521E-15	1.930E-15	1.983E-17
0.6	1.00E-10	5.238E-15	2.922E-15	4.669E-17
0.8	2.20E-9	7.278E-15	3.863E-15	1.232E-17
1.0	2.49E-8	3.088E-14	4.866E-15	1.268E-17

Example 4 *[31] The next problem is*

$$u'(x) = 1 - \int_0^x u(t)\,dt$$

with the initial condition $u(0) = 0$.

The exact solution of the problem is $u(x) = \sin x$. We applied the present method for $N = 5, 10, 12, 14$. The maximum error for $N = 5$ in the Taylor method [31] is $0.195E-3$, and it is $9.666E-6$ in the Bessel collocation method [34], while it is $7.124E-7$ in our method with the same N value. The errors obtained by the Tau method [22] with $N = 10$ and errors of our solutions obtained with $N = 10, 12, 14$ are presented in Table 4. It is seen that our method is superior to the Tau method for $N = 10$, and our solutions are improved significantly by increasing the value of N. Lastly, the system matrix of this problem has condition number equal to 1.6598, 2.6977, 2.4255, and 1.6527 for $N = 5, 10, 12$, and 14, respectively.

5. Conclusions

In this paper, we introduce an operational matrix method based on the operational matrices of integration and product for Taylor polynomials. The least-squares approximation is used to obtain the required operational matrices. By using operational matrices without collocation points, we convert the considered integro-differential equation problem to a system of algebraic equations. Numerous other matrix methods use the operational matrix of differentiation to establish relations between the unknown function and its derivatives [1, 31, 34]. In our case, we use an integration matrix instead. Numerical results show that the accuracy of our method is superior to that of the homotopy perturbation method [33], Bernoulli polynomials [5], Tau method [22] and the Bessel collocation method [34] with collocation points. Our method yields good approximations of the solutions of the considered problems even for small values of N. On the whole, we can comment that the numerical scheme presented in this study is an easy-to-implement method that can be relied on to solve problems of similar type with a reasonable level of accuracy.

Acknowledgment

The authors are supported by the Scientific Research Project Administration of Akdeniz University.

References

[1] Akyüz-Daşcıoğlu A. A Chebyshev polynomial approach for linear Fredholm-Volterra integro-differential equations in the most general form. Appl Math Comput 2006; 181: 103-112.

[2] Alexander C, Sadiku M. Fundamentals of Electric Circuits. available at http://www.coe.ufrj.br/˜ julio/Fundamentals

[3] Ali I. Jacobi-spectral methods for integro-delay differential equations with weakly singular kernels. Turk J Math 2015; 39: 810-819.

[4] Aslan B, Gürbüz B, Sezer M. A new collocation method for solution of mixed linear integro-differential difference equations. NTMSCI 2015; 3: 133-146.

[5] Bhrawy AH, Tohidi E, Soleymani F. A new Bernoulli matrix method for solving high-order linear and nonlinear Fredholm integro-differential equations with piecewise intervals. Appl Math Comput 2012; 219: 482-497.

[6] Borhanifar A, Sadri KH. A new operational approach for numerical solution of generalized functional integro-differential equations. J Comput Appl Math 2015; 279: 80-96.

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

[8] Danfu H, Xufeng S. Numerical solution of integro-differential equations by using CAS wavelet operational matrix of integration. Appl Math Comput 2007; 194: 460-466.

[9] Darania P, Ebadian A. A method for the numerical solution of the integro-differential equations. Appl Math Comput 2007; 188: 657-668.

[10] Doha EH, Abd-Elhameed WM, Youssri YH. Second kind Chebyshev operational matrix algorithm for solving differential equations of Lane-Emden type. New Astron 2013; 23: 113-117.

[11] Hosseini SM, Shahmorad S. Numerical solution of a class of integro-differential equations by the Tau method with an error estimation. Appl Math Comput 2003; 136: 559-570.

[12] Hsiao C. Hybrid function method for solving Fredholm and Volterra integral equations of the second kind. J Comput Appl Math 2009; 230: 59-68.

[13] Kajani M, Ghasemi M, Babolian E. Numerical solution of linear integro-differential equation by using sine-cosine wavelets. Appl Math Comput 2006; 180: 569-574.

[14] Khan Y, Fardi M. A new efficient multi-parametric homotopy approach for two-dimensional Fredholm integral equations of the second kind. Hacet J Math Stat 2015; 44: 93-99.

[15] Khan Y, Sayevand K, Fardi M, Ghasemi M. A novel computing multi-parametric homotopy approach for system of linear and nonlinear Fredholm integral equations. Appl Math Comput 2014; 249: 229-236.

[16] Kress R. Linear Integral Equations. New York, NY, USA: Springer Science Business Media, 2014.

[17] Kreyszig E. Introductory Functional Analysis with Applications. New York, NY, USA: Wiley, 1989.

[18] Maleknejad K, Mirzaee F. Numerical solution of integro-differential equations by using rationalized Haar functions method. Kybernetes 2006; 35: 1735-1744.

[19] Maleknejad K, Basirat B, Hashemizadeh E. A Bernstein operational matrix approach for solving a system of high order linear Volterra-Fredholm integro-differential equations. Math Comput Model 2012; 55: 1363-1372.

[20] Razzaghi M, Yousefi S. Legendre wavelets method for the nonlinear Volterra-Fredholm integral equations. Math Comput Simulat 2005; 70: 1-8.

[21] Saadatmandi A, Dehghan M. Numerical solution of the higher-order linear Fredholm integro-differential-difference equation with variable coefficients. Comput Math Appl 2010; 59: 2996-3004.

[22] Shahmorad S. Numerical solution of the general form linear Fredholm-Volterra integro-differential equations by the Tau method with an error estimation. Appl Math Comput 2005; 167: 1418-1429.

[23] Smarda Z, Khan Y. Singular initial value problem for a system of integro-differential equations unsolved with respect to the derivative. Appl Math Comput 2013; 222: 290-296.

[24] Stenger F. Numerical Methods Based on Sinc and Analytic Functions. New York, NY, USA: Springer, 1993.

[25] Tadmor E, Athavale P. Multiscale image representation using integro-differential equations. Inverse Probl Imag 2009; 3: 693-710.

[26] Tikhonov A, Arsenin V. Solutions of ill-posed problems. Washington, DC, USA: V. H. Winston & Sons, 1977.

[27] Vanani S, Aminataei A. Numerical solution of Volterra integro-differential equations. J Comput Anal Appl 2011; 13: 654-662.

[28] Wazwaz AM. Linear and Nonlinear Integral Equations. New York, NY, USA: Springer Science & Business Media, 2011.

[29] Wazwaz AM. The combined Laplace transform-Adomian decomposition method for handling nonlinear Volterra integro-differential equations. Appl Math Comput 2010; 216: 1304-1309.

[30] Xue D, Chen Y. Scientific computing with MATLAB, 2nd edition. Boca Raton, FL, USA: CRC Press, 2016.

[31] Yalçınbaş S, Sezer M. The approximate solution of high-order linear Volterra-Fredholm integro-differential equations in terms of Taylor polynomials. Appl Math Comput 2000; 112: 291-308.

[32] Yousefi SA, Behroozifar M. Operational matrices of Bernstein polynomials and their applications. Int J Syst Sci 2010; 41: 709-716.

[33] Yusufoğlu E. Improved homotopy perturbation method for solving Fredholm type integro-differential equations. Chaos Soliton Fract 2009; 41: 28-37.

[34] Yüzbaşı Ş, Şahin N, Yıldırım A. A collocation approach for solving high-order linear Fredholm-Volterra integro-differential equations. Math Comput Model 2012; 55: 547-563.

[35] Zhao J, Corless RM. Compact finite difference method for integro-differential equations. Appl Math Comput 2006; 177: 271-288.