UJM 13 (2) 2024
UNNES Journal of Mathematics
http://journal.unnes.ac.id/sju/index.php/ujm

Lucas's Matrix Approach in Solving First Order Linear Volterra Integro-Differential Equations

Kamoh, N. M. [1]*, Dang, B. C. [2], Soomiyol, M. C. [3]

[1*,2]Department of Mathematics, University of Jos, Nigeria
[3]Department of Mathematics, Benue State University, Makurdi, Nigeria

Article Info

Article History:
Recived ....
Accepted .....
Published .....

Keywords:
Volterra integro-differential equations; matrix of derivative first Order equations; Lucas's polynomials function.

Abstract

In this paper, matrix calculus of the Lucas polynomials is derived for the numerical solution of first order linear Volterra integro-differential equations. The equation is solved by transforming the differential part of the equation using the Lucas polynomials matrix of derivatives and the integral part is evaluated base on the Lucas polynomial's function. The new method possesses the desirable feature of being a strong and dependable technique for solving many Volterra integro-differential equations of the first order. The developed technique was illustrated on some test problems in literature and results confirmed that the developed technique is more accurate than those developed by some considered authors. The present work suggests that the proposed scheme is comparatively simpler to apply than many other existing methods, whereas the numerical results revealed the accuracy and superiority of the presented method. The prime attraction of the present technique is displayed by the comparative study. The superior results for different input values suggest the novelty of the present work and it is worthy to solve the considered type equations approximately.

.

How to cite:

Kamoh, N. M., Dang, B.C., & Soomiyol, M.C. 2024. Lucas's Matrix Approach in Solving First Order Linear Volterra Integro-Differential Equations. *UNNES Journal of Mathematics*. 13 (2): 20-28.

*Correspondence Address
E-mail: mahwash1477@gmail.com

p-ISSN 2252-6943
e-ISSN 2460-5859

1. Introduction

The history of Volterra integro differential equations appear quite early in the mathematical development of theoretical population dynamics in the pioneering work of such mathematicians as Volterra and Kostitzin. In their attempts to model the growth of populations by means of differential equations these early investigators were quick to point out that the current growth rate of a population is unlikely to depend only on the current population size or that growth rates are unlikely to respond instantaneously to changes in population sizes or densities. This led Volterra in particular to include functions of Volterra integral type in what have become the classical differential models of population dynamics and mathematical ecology (equations such as the logistic equation, the famous predator-prey system of Volterra and the well-known Volterra-Lotka competition model). Much of this early work involving integro differential equations in population dynamics can be found in the collection of Volterra (1959) and Wazwaz (2011). This equation arises in the modeling of natural systems where the past influences the present and future, for example pollution, population growth, mechanical systems and financial market. It is characterized by the existence of one or more of the derivatives $y'(x), y''(x)$, outside the integral sign. These equations may be observed when we convert an initial value problem to an integral equation by using Leibnitz rule. The general form of a linear Volterra integro-differential equation (VIDE) of the first order is given as;

$$y'(x) = f(x) + \lambda \int_{x_0}^{x} \mathcal{H}(x,t)y(t)dt, \qquad x \epsilon [x_0, X] \qquad (1)$$

with initial conditions $y(0) = y_0$, λ is a constant, $\mathcal{H}(x,t)$ is the kernel or nucleus and $y(x)$ is the unknown function to be determined.

A variety of powerful methods has been presented, such as the homotopy analysis method (Wazwaz, 2011), homotopy perturbation method (Ghasemi *et al.,* 2006; Wazwaz, 2011), operational matrix with Block-Pulse functions method (Babolian *et al*., 2008), variational iteration method (Salih *et al.,* 2013; Wazwaz 2011), and the Adomian decomposition method (El-Sayed *et al.,* 2004;Jerri, 1999; Wazwaz, 2011). Some fundamental works on various aspects of modifications of the Adomian's decomposition method are given by Araghi & Behzadi (2009). Integro-Differential Equations (IDEs) play an important role in many branches of linear and nonlinear functional analysis and their applications are found in the theory of engineering, mechanics, physics, chemistry, astronomy, biology and economics (Dadkhah*, et al.,* 2010; Ebadi, *et al.,* 2009; Gherjalar & Mohammadikia, 2012).

Unfortunately, only very few of these problems can be solved analytically. An analytic treatment of the nonlinear Volterra integro–differential equations was developed using a combined form of the Laplace transform method with the Adomian decomposition by Wazwaz (2010). Kamoh & Aboiyar (2018) introduced continuous linear multistep method for the general solution of first order initial value problems for volterra integro-differential equations. Series solution for weakly-Singular Kernel VIDEs was obtained suing a combined form of the Laplace transform method with the Adomian decomposition by Soori (2016). The nth-order integro-differential equations were solved using a combined form of the Laplace transform method with the Adomian decomposition by Al-Hayani (2013). However, the numerical solutions of such equations have been extensively studied by many researchers. A practical matrix method, which is based on collocation points was developed to find approximate solutions of high-order linear Volterra integro-differential equations (VIDEs) using Bessel polynomials method (Sezer *et al.,* 2011). The application of orthonormal Bernstein polynomials to construct an efficient scheme for solving fractional stochastic integro-differential equation was suggested Farshid & Nasrin (2017).

A numerical method for nonlinear Volterra integro-differential equations was introduced in Feldstein & Sopka (1974). The Laguerre approach for solving pantograph-type Volterra integro-differential equations was introduced Yüzbasi (2014). The implicit Runge-Kutta methods of optimal order for Volterra integro-differential equations were suggested by Brunner (1984). Solving nonlinear fractional integro-differential equations of Volterra type using novel mathematical matrices was discussed Farshid *et al.,* (2015). The spectral method for Volterra functional integro-differential equations of neutral type was treated Sedaghat *et al.,* (2014). The mixed interpolation and collocation methods for first and second order Volterra integro-differential equations with periodic solution were introduced in Brunner *et al.,* (1997). Kamoh *et al.,* (2017) presented a continuous multistep method for volterra integro-differential equations of the second order.

The numerical solution of a non-linear Volterra integro-differential equation via Runge-Kutta-Verner method was developed Filiz (2013). The Legendre spectral collocation method for neutral and high-order Volterra integro-differential equation was investigated Wei & Chen, (2014). A numerical framework for solving high-order pantograph-delay Volterra integro-differential equations and the numerical solution of optimal control problem of the non-linear Volterra integral equations via generalized hat functions were respectively discussed by Farshid *et al.,* (2016*)* and Farshid & Elham (2016). Approximate solution of high order linear Volterra-Fredholm integro-differential equations in terms of Taylor polynomials was considered Yalcinbas & Sezer (2000). The quadrature rules to find the numerical solutions of the initial value problems for Volterra integro-differential equations of the second kind appeared in Atifa (2003). Linear multistep method for Volterra integro-differential equations was constructed by Day (1967) and Linz (1969). An improved method based on Haar wavelets for the numerical solution of nonlinear integral and integro-differential equations of first and higher orders and Numerical solution of integro-differential equations by using rationalized Haar functions methods was developed Siraj-ul-Islam *et al.,* (2014) and Maleknejed & Mirzaee (2006) respectively. Numerical solutions of systems of high-order Fredholm integro-differential equations using Euler polynomials were suggested Farshid & Saeed (2015). The Numerical solution of the system of nonlinear Volterra integro-differential equations with nonlinear differential part by the operational Tau method and error estimation was suggested Abbasbandy & Taati, (2009).

In this work, our main objective is to produce a simpler and efficient methodology to solve the considered class of equations. For that purpose, we have adopted Lucas's polynomial basis function for solving the first-order Volterra integro-differential equation.

2. Method

The Lucas polynomial function is of the form;

$$y_N(x) = \sum_{i=0}^{N} \gamma_i \, \delta_i(x) \tag{2}$$

where $\gamma_i \in \mathbb{R}$, $y \epsilon C^1(a,b)$ can be expressed in matrix form as shown below

$$y_{\mathrm{N}}(x) \cong \sum_{i=0}^{N} \gamma_i \, \delta_i(x) = \mathcal{L}(x)\Omega^T \tag{3}$$

Where the Lucas coefficients Ω vector and the Lucas vector $\mathcal{L}(x)$ are given by

$$\left.\begin{aligned} \Omega &= [\gamma_0, \gamma_1, \gamma_2, \dots, \gamma_{\mathrm{N}}] \\ \mathcal{L}(x) &= [\delta_0(x), \delta_1(x), \dots, \delta_{\mathrm{N}}(x)] \end{aligned}\right\} \tag{4}$$

Introducing the Lucas vector $\mathcal{L}(x)$ in the form $\mathcal{L}(x) = [\delta_0(x), \delta_1(x), \dots, \delta_{\mathrm{N}}(x)]$, the derivative of the Lucas vector $\mathcal{L}(x)$ can be expressed in matrix form as

$$(\mathcal{L}(x)')^T = \phi \mathcal{L}(x)^T = \mathcal{L}(x)(\phi)^T \tag{5}$$

where

$$(\mathcal{L}(x)')^T = \begin{pmatrix} \delta_0{}'(x) \\ \delta_1{}'(x) \\ \delta_2{}'(x) \\ \vdots \\ . \\ \delta_{\mathrm{N}}{}'(x) \end{pmatrix}, \ \phi = \begin{pmatrix} 0 & \frac{1}{2} & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 2 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & \vdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \vdots & 0 & 0 & 0 \\ . & . & . & \dots & . & . & . \\ . & . & . & & . & (\mathrm{N}-1) & . \\ 0 & 6 & 0 & \dots & 0 & 0 & 0 \end{pmatrix} \text{ and } \mathcal{L}(x)^T = \begin{pmatrix} \delta_0(x) \\ \delta_1(x) \\ \delta_2(x) \\ \vdots \\ . \\ \delta_{\mathrm{N}}(x) \end{pmatrix}$$

Where the matrix ϕ is an $(\mathrm{N}+1) \times (\mathrm{N}+1)$ operational matrix of derivatives, similarly, the k^{th} derivative of $L(x)$ can be obtained from the following relation for any positive integer $k \in \mathbb{N}$ as;

$$\frac{d\mathcal{L}(x)}{dx} = D^{(1)}\mathcal{L}(x), \frac{d^2\mathcal{L}(x)}{dx^2} = D^{(2)}\mathcal{L}(x), \dots, \frac{d^k\mathcal{L}(x)}{dx^k} = D^{(k)}\mathcal{L}(x) \tag{6}$$

In this paper, we shall use the collocation method based on Lucas operational matrix of derivatives to solve numerically the Volterra type first order linear integro-differential equation with initial conditions.

The approximate solution is investigated under the consideration that the solution of considered class of equation is analytic on the prescribed domain. We assume that approximate solution of (1) is of the following form (2), where $\delta_i(x)'s$, are monomial bases and $\gamma_i's$ are real coefficients to be determined. Implementing (2) and its derivatives on both sides of equation (1) we have the following estimates,

$$\sum_{i=0}^{N} \gamma_i \, \delta_i(x_j)' = \mathcal{L}(x_j)(\phi^T)^{(1)}\Omega^T = f\,(x) + \int_{x_0}^{x_j} \mathcal{H}(x_j, t) \sum_{i=0}^{N} \gamma_i \, \delta_i(t) dt \tag{7}$$

where Ω represent Lucas's coefficients vector, $\mathcal{L}(x)$ is the Lucas vector and ϕ is the matrix of derivatives defined in (5). Afterwards, collocation method is used at the points $x_j = \frac{j}{\mathrm{N}-1}$, $j = 0,1,2, \dots, \mathrm{N}-1$.

The system (7) is written in matrix notation as,

$$JK = V \tag{8}$$

where

$$\left.\begin{aligned} J &= \begin{pmatrix} \delta_0(x_0) \\ \delta_1(x_1) \\ \delta_2(x_2) \\ \vdots \\ . \\ \delta_{\mathrm{N}}(x_N) \end{pmatrix}^T \begin{pmatrix} \delta_0{}'(x_0) & \delta_1{}'(x_0) & \cdots & \delta_{\mathrm{N}}{}'(x_0) \\ \delta_0{}'(x_1) & \delta_1{}'(x_1) & \vdots & \delta_{\mathrm{N}}{}'(x_1) \\ \vdots & \vdots & \vdots & \vdots \\ \delta_0{}'(x_N) & \delta_1{}'(x_N) & \cdots & \delta_{\mathrm{N}}{}'(x_N) \end{pmatrix}, K = \begin{pmatrix} \gamma_0 \\ \gamma_1 \\ \gamma_2 \\ \vdots \\ . \\ \gamma_{\mathrm{N}} \end{pmatrix} \\ &\text{and} \\ V &= f\left(x_j, y(x_j), \int_{x_0}^{x_j} \mathcal{H}(x_j, \mathrm{t}) \sum_{i=0}^{N} \gamma_i \, \delta_i(t) dt\right), \ x_j = \frac{j}{\mathrm{N}-1}, j = 0,1, \dots, \mathrm{N}-1 \end{aligned}\right\} \tag{9}$$

The consistency of (8) is assured since $rank\,(J) = rank\,(J|V) = N + 1$ and subsequently K is determined. Finally, the approximate solution follows from (8) by Newton's iteration method.

3. Results and discussions

3.1 Bound of error estimation

In this part, we develop the residual error and the absolute error methods. On substituting the approximate solution in equation (8), the resulting relation is obtained as,

$$\mho_T(x_j) = y'(x_j) - \int_{x_0}^{x_j} \mathcal{H}(x_j, \mathrm{t}) y(t) dt - V(x_j) \qquad (10)$$

where x_j's are given by $j = 0,1,2,\dots N-1$. Consider $\mho_T(x_j) \leq 10^{\alpha j}$ such that αj is any positive number. If maximum of $10^{-\alpha j} = 10^{-\alpha}$, then the truncation limit T is increased until $\mho_T(x_j)$ at each point is smaller than the desired accuracy $10^{-\alpha}$. For sufficiently large T; $\mho_T(x_j) \to 0$; in that case, error decreases rapidly. However, the boundedness of residual function can be estimated by adopting standard inequality from integral calculus, we have

$$\left| \int_{x_0}^{x} \mho_T(t) dt \right| \leq \int_{x_0}^{x} |\mho_T(t)| dt,$$

and implementing integral mean value theorem, the upper bound of mean error is obtained as

$$\mho_T(\theta) \leq \frac{\int_{x_0}^{x} |\mho_T(t)| dt}{x - x_0}; \; x_0 \leq \theta \leq x.$$

3.2 Numerical illustrations and results

We now implement the derived method on some initial value problems of the Volterra type integro-differential equations to support the theoretical discussion. The proposed method is tested on some numerical examples contained in literature using Scientific Workplace 5.5. The absolute error is defined by $|y(x_j) - y_j|, j = 0, 1, 2, \dots, N$, (11) where $y(x_j)$ and y_j stands for exact and approximate solution at $x = x_j$, respectively.

Example 1
Consider a first order Linear Volterra integro-differential equation of the form

$$y'(x) = 1 - \int_0^x y(t) dt\,, y(0) = 0, \qquad 0 \leq x \leq 1$$

The exact solution is $y(x) = sinx$ (Day, 1967). Table 1 shows the comparison of the exact and approximate solution. Applying the proposed technique with the initial conditions by collocating at the points $x_j = \frac{j}{\mathrm{N}-1}$, $j = 0,1,2,\dots,\mathrm{N}-1$, with $\mathrm{N} = 10$ and solving the resulting algebraic systems of equations, we have the following unknown coefficients.

$$\gamma_0 = 4.057068938\,66 \times 10^{-6}, \gamma_1 = 1.590652193\,85, \gamma_2 = -7.418477646\,5 \times 10^{-6},$$

$$\gamma_3 = -0.212750306\,657, \gamma_4 = 5.412618263\,59 \times 10^{-6}, \gamma_5 = 9.830111503\,27 \times 10^{-3},$$

$$\gamma_6 = -2.824400220\,43 \times 10^{-6}, \gamma_7 = -2.256274008\,61 \times 10^{-4}, \gamma_8 = 9.046972734\,51 \times 10^{-7},$$

$$\gamma_9 = 3.062268304\,98 \times 10^{-6}, \gamma_{10} = -1.315066087\,69 \times 10^{-7}$$

Substituting these values into (7.0) gives the approximate solution to the problem as

$$\begin{aligned} y_{10}(x) = {} & -1.315066087\,69 \times 10^{-7} x^{10} + 3.062268304\,98 \times 10^{-6} x^9 - 4.103688142\,39 \times 10^{-7} x^8 - 1.980669861\,16 \times 10^{-4} x^7 \\ & - 1.895533397\,37 \times 10^{-7} x^6 + 8.333400941\,48 \times 10^{-3} x^5 - 1.516802842 \times 10^{-8} x^4 - 0.166666664\,704 x^3 \\ & - 1.15420019 \times 10^{-10} x^2 + 1.0x + 4.000000000\,00 \times 10^{-18} \end{aligned}$$

Table 1 Comparison of error estimation result

x	Exact Solution	Error estimation by Proposed Method For $N = 10$	Error estimation of LMM Method by Kamoh & Aboiyar (2018)	Error estimation of the numerical method by Day (1967)
0.1	0.0998334166468	1.91×10^{-13}	1.419×10^{-7}	3.840×10^{-5}
0.2	0.198669330 795	1.63×10^{-13}	1.108×10^{-6}	1.660×10^{-4}
0.3	0.295520206 661	1.69×10^{-13}	3.747×10^{-6}	2.440×10^{-4}
0.4	0.389418342 309	1.69×10^{-13}	8.741×10^{-6}	3.160×10^{-4}
0.5	0.479425538 604	1.73×10^{-13}	1.693×10^{-5}	3.800×10^{-4}
0.6	0.564642473 395	1.82×10^{-13}	2.892×10^{-5}	4.330×10^{-4}
0.7	0.644217687 238	1.94×10^{-13}	4.532×10^{-5}	4.740×10^{-4}
0.8	0.717356090 900	2.20×10^{-13}	6.666×10^{-5}	4.990×10^{-4}
0.9	0.783326909 627	4.39×10^{-13}	9.324×10^{-5}	5.090×10^{-4}
1.0	0.841470984 808	4.39×10^{-13}	1.254×10^{-4}	5.020×10^{-4}

Examples 2

Consider the first order initial value problem (IVP) of the Volterra type integro-differential equation;

$$y'(x) = -sinx - cosx + \int_0^x 2cos(x-t)y(t)dt, \quad y(0) = 1, 0 \leq x \leq 1$$

The exact solution is $y(x) = e^{-x}$ (Nur *et al.*, 2018). Table 2 shows the comparison of the exact and approximate solution.

Applying the proposed technique with the initial conditions by collocating at the points $x_j = \frac{j}{\mathrm{N}-1}$, $j = 0,1,2,\dots,\mathrm{N}-1$ with $\mathrm{N} = 10$ and solving the resulting algebraic systems of equations, we have the following unknown coefficients

$$\gamma_0 = 0.111948654\,681, \gamma_1 = -0.576714853\,149, \gamma_2 = 0.352828136\,232,$$

$$\gamma_3 = -0.128949868\,217, \gamma_4 = 3.399988449\,01 \times 10^{-2}, \gamma_5 = -7.036906711\,29 \times 10^{-3}$$

$$\gamma_6 = 1.200339873\,26 \times 10^{-3}, \qquad \gamma_7 = -1.754748555\,07 \times 10^{-4}$$

$$\gamma_8 = 2.281680519\,62 \times 10^{-5}, \qquad \gamma_9 = -2.520517163\,03 \times 10^{-6}$$

$$\gamma_{10} = 1.679193302\,17 \times 10^{-7}$$

Substituting these values into (7) gives the approximate solution to the problem as

$$\begin{aligned} y_{10}(x) = {} & 1.679193302\,17 \times 10^{-7}x^{10} - 2.520517163\,03 \times 10^{-6}x^9 + 2.449599849\,84 \times 10^{-5}x^8 - 1.981595099\,74 \times 10^{-4}x^7 \\ & + 1.388751491\,39 \times 10^{-3}x^6 - 8.333284663\,24 \times 10^{-3}x^5 + 4.166665580\,01 \times 10^{-2}x^4 \\ & - 0.166666665\,265x^3 + 0.499999999\,918x^2 - 1.000000000\,00x + 1.0 \end{aligned}$$

Table 2 Comparison of numerical results of example 2 for $h = 0.1$

x	Exact Solution	Approximate Solution of the proposed method	Error estimation of Proposed Method For $N = 10$
0.1	0.904837418 036	0.904837418 036	1.33199×10^{-13}
0.2	0.818730753 078	0.818730753 078	1.03768×10^{-13}
0.3	0.740818220 682	0.740818220 682	8.86683×10^{-14}
0.4	0.670320046 036	0.670320046 036	5.39836×10^{-14}
0.5	0.606530659 713	0.606530659 713	4.35548×10^{-15}
0.6	0.548811636 094	0.548811636 094	6.52532×10^{-14}
0.7	0.496585303 791	0.496585303 792	1.59860×10^{-13}
0.8	0.449328964 117	0.449328964 117	2.74944×10^{-13}
0.9	0.406569659 741	0.406569659 741	4.42679×10^{-13}
1.0	0.367879441 171	0.367879441 172	4.99270×10^{-13}

Example 3
Consider the first order initial value problem (IVP) of the Volterra type integro-differential equation;

$$y'(x) = -y + \int_0^x e^{(t-x)} y(t)dt, \; y(0) = 1, 0 \le x \le 1$$

The exact solution is $y(x) = e^{-x} coshx$ (Raftari, 2010). Table 3 shows the comparison of the exact and approximate solution.

Applying the proposed technique with the initial conditions by collocating at the points $x_j = \frac{j}{N-1}$, $j = 0,1,2, \dots, N-1$ for $N = 10$ and solving the resulting algebraic systems of equations, we have the following unknown coefficients

$$c_0 = 0.153321340\,355, c_1 = 3.851933741\,86 \times 10^{-2}, c_2 = 0.177439830\,832, c_3 = -0.218624902\,635, c_4 = 0.143687229\,192, c_5 = -6.472260379\,80 \times 10^{-2},$$

$$c_6 = 2.307017851\,84 \times 10^{-2}, c_7 = -7.751163453\,92 \times 10^{-3}, c_8 = 2.428437087\,37 \times 10^{-3}, c_9 = -5.303851495\,71 \times 10^{-4}, c_{10} = 5.298401586\,55 \times 10^{-5}$$

Substituting these values into (7.) gives the approximate solution to the problem as

$$y_{10}(x) = 5.298401586\,55 \times 10^{-5}x^{10} - 5.303851495\,71 \times 10^{-4}x^9 + 2.958277246\,03 \times 10^{-3}x^8 - 1.252462980\,01 \times 10^{-2}x^7 + 4.435211577\,27 \times 10^{-2}x^6 - 0.133301147\,014x^5 + 0.333326242\,843x^4 - 0.666665764\,467x^3 + 0.999999948\,06x^2 - 1.000000000\,00x + 1.0$$

Table 3 Comparison of numerical results of example 3

x	Exact Solution	Approximate solution by proposed method with $N = 10$	Approx solution by Homotopy Perturbation Method Raftari (2010) with $N = 10$	Approx solution by Finite difference method Raftari (2010) with $N = 12$
0.0833	0.923269079443	0.923269079 366	0.9232408643	0.9409611903
0.1667	0.858241771706	0.858241771 637	0.8582656556	0.8560967849
0.2500	0.803265329856	0.803265329 796	0.8032653310	0.8221926661
0.3333	0.756725673 991	0.756725673 931	0.7567085596	0.7521848166
0.4167	0.717284618129	0.717284618 075	0.7172991016	0.7379172201
0.5000	0.683939720586	0.683939720533	0.6839397209	0.6768034806
0.5833	0.655711992411	0.655711992360	0.6557016132	0.6667601409
0.6667	0.631789782779	0.631789782732	0.6317985685	0.6235899136
0.7500	0.611565080074	0.611565080025	0.6115650814	0.6149784301
0.8333	0.594444097482	0.594444097438	0.5944378021	0.5862745171
0.9167	0.579934543893	0.579934543852	0.5799398782	0.5770459174
1.0000	0.567667641618	0.567667641507	0.5676676511	0.5643959592

Example 4
Consider the Volterra integro-differential equation

$$y'(x) = (x^2 + 2x + 1)e^{-x} + 5x^2 + 8 - y - \int_0^x ty(t)dt, \; 0 \le x \le 1, y(0) = 10$$

which has the exact solution $y(x) = 10 - xe^{-x}$. The numerical results are represented in Table 4 (Raftari, 2010)

Applying the proposed technique with the initial conditions by collocating at the points $x_j = \frac{j}{N-1}$, $j = 0,1,2, \dots, N-2$ for $N = 4$ and solving the resulting algebraic systems of equations, we have the following unknown coefficients

$$c_0 = 4.423310723\,75, c_1 = 0.129046552\,457, c_2 = 0.447717617\,895,$$

$$c_3 = -0.318906784\,000, c_4 = 0.121948501\,721, c_5 = -3.276535961\,05 \times 10^{-2},$$

$$c_6 = 6.842359903\,88 \times 10^{-3}, c_7 = -1.185529048\,7 \times 10^{-3},$$

$$c_8 = 1.791981195\,23 \times 10^{-4}, c_9 = -2.229989597\,52 \times 10^{-5},$$

$$c_{10} = 1.598613622\,78 \times 10^{-6}$$

Substituting these values into (7) gives the approximate solution to the problem as

$$y_{10}(x) = 1.598613622\,78 \times 10^{-6}x^{10} - 2.229989597\,52 \times 10^{-5}x^9 + 1.951842557\,51 \times 10^{-4}x^8 - 1.386228112\,48 \times 10^{-3}x^7 + 8.331896336\,86 \times 10^{-3}x^6 - 4.166616014\,27 \times 10^{-2}x^5 + 0.166666554\,216x^4 - 0.499999985\,614x^3 + 0.999999999\,167x^2 - 1.0x + 10.0$$

Table 4 Comparison of numerical results of example 4

x	Exact Solution	Proposed method with $N = 10$	Finite Difference method by Raftari (2010) with $N = 12$	HPM method by Raftari (2010) with $N = 4$
0.0714	9.933520218 13	9.933520218 13	9.933780913	9.933495
0.1428	9.876202761 99	9.876202761 99	9.876458584	9.876160
0.2142	9.827100559 27	9.827100559 27	9.827589590	9.827049
0.2857	9.785299870 25	9.785299870 24	9.785803615	9.785307
0.3571	9.750136228 69	9.750136228 68	9.750832589	9.750163
0.4286	9.720801197 16	9.720801197 16	9.721466032	9.720923
0.5000	9.696734670 14	9.696734670 14	9.697552931	9.696975
0.5714	9.677310845 53	9.677310845 52	9.678042251	9.677777
0.6428	9.662004144 4	9.662004144 4	9.662857435	9.662846
0.7143	9.650325387 97	9.650325387 96	9.651100634	9.651781
0.7857	9.641877523 80	9.641877523 80	9.642739377	9.644241
0.8571	9.636254445 21	9.636254445 21	9.637018786	9.639963
0.9286	9.633104129 72	9.633104129 72	9.633929509	9.638737
1.000	9.632120558 83	9.632120558 82	9.632846912	9.640444

4 Conclusion

In this research, we proposed a new approach for solving first order linear Volterra integro differential equations and comparisons were made with the existing method. The present work suggests that the proposed scheme is comparatively simpler to apply than many other existing methods, whereas the numerical results revealed the accuracy and superiority of the presented method. The prime attraction of the present technique is displayed by the comparative study. The superior results for different input values suggest the novelty of the present work and it is worthy to solve the considered type equations approximately.

References

Akinola, E. I., Olopade, I. A., Akinpelu, F. O., Areo, A. O., & Oyewumi, A. A. (2016). Sumudu transform series decomposition method for solving nonlinear Volterra integro-differential equations. *Innovative Space of Scientific Research Journals, 18*(1), 262–267.

Araghi, M. A., & Behzadi, S. S. (2009). Solving nonlinear Volterra–Fredholm integro-differential equations using the modified Adomian decomposition method. *Computational Methods in Applied Mathematics, 9*, 321–331.

Atifa, A. (2003). *Approximated methods for first order Volterra integro-differential equations* (M.Sc. thesis, University Technology).

Babolian, E., Masouri, Z., & Hatamzadeh, S. (2008). New direct method to solve nonlinear Volterra–Fredholm integral and integro-differential equations using operational matrix with block-pulse functions. *Progress in Electromagnetics Research B, 8*, 59–76.

Baharum, N. A., Abdul Majid, Z., & Senu, N. (2018). Solving Volterra integro-differential equations via diagonally implicit multistep block method. *International Journal of Mathematics and Mathematical Sciences, 2018*, Article ID 7392452, 10 pages.

Brunner, H. (1984). Implicit Runge–Kutta methods of optimal order for Volterra integro-differential equations. *Mathematics of Computation, 42*(165), 95–109.

Brunner, H., Makroglou, A., & Miller, R. K. (1997). The mixed interpolation and collocation methods for first and second order Volterra integro-differential equations with periodic solution. *Applied Numerical Mathematics, 23*(4), 381–402.

Chen, H., & Zhang, C. (2011). Boundary value methods for Volterra integral and integro-differential equations. *Applied Mathematics and Computation, 218*(6), 2619–2630.

Dadkhah, M., Kajani, M. T., & Mahdavi, S. (2010). Numerical solution of non-linear Fredholm–Volterra integro-differential equations using Legendre wavelets. *Proceedings of the International Conference on Mathematics and Applications (ICMA)*.

Day, J. T. (1967). Note on the numerical solution of integro-differential equations. *Journal of Computation, 9*, 394–395.

Ebadi, G., Rahimi-Ardabili, M. Y., & Shahmorad, S. (2009). Numerical solution of the system of nonlinear Volterra integro-differential equations. *Southeast Asian Bulletin of Mathematics, 33*, 835–846.

El-Sayed, S. M., Kaya, D., & Zarea, S. (2004). The decomposition method applied to solve high-order linear Volterra–Fredholm integro-differential equations. *International Journal of Nonlinear Sciences and Numerical Simulation, 5*(2), 105–112.

Fadhel, F. S., Mezaal, A. O., & Salih, S. H. (2013). Approximate solution of the linear mixed Volterra–Fredholm integro-differential equations of the second kind by using variational iteration method. *Al-Mustansiriyah Journal of Science, 24*(5), 137–146.

Feldstein, A., & Sopka, J. R. (1974). Numerical methods for non-linear Volterra integro-differential equations. *SIAM Journal on Numerical Analysis, 11*, 826–846.

Ghasemi, M., Kajani, M., & Babolian, E. (2007). Application of He’s homotopy perturbation method to nonlinear integro-differential equations. *Applied Mathematics and Computation, 188*, 538–548.

Gherjalar, H. D., & Mohammadikia, H. (2012). Numerical solution of functional integral and integro-differential equations by using B-splines. *Applied Mathematics, 3*, 1940–1944.

Jerri, A. M. (1999). *Introduction to integral equations with applications*. New York: Wiley.

Kamoh, N. M., & Aboiyar, T. (2018). Continuous linear multistep method for the general solution of first-order initial value problems for Volterra integro-differential equations. *Multidiscipline Modeling in Materials and Structures, 14*(5), 960–969.

Kamoh, N. M., Aboiyar, T., & Kimbir, A. R. (2017). Continuous multistep methods for Volterra integro-differential equations of the second order. *Science World Journal, 12*(3), 11–14.

Lambert, J. D. (1973). *Computational methods in ordinary differential equations*. John Wiley, New York.

Linz, P. (1969). Linear multistep methods for Volterra integro-differential equations. *Journal of the Association for Computing Machinery, 16*(2), 295–301.

Polishuk, Ye. (1977). *M. Vito Volterra*. Leningrad: Nauka.

Pour-Mahmoud, J., Rahimi-Ardabili, M. Y., & Shahmorad, S. (2005). Numerical solution of Volterra integro-differential equations by the Tau method with the Chebyshev and Legendre bases. *Applied Mathematics and Computation, 170*, 314–338.

Sachin, B., & Jayvant, P. (2016). A novel third order numerical method for solving Volterra integro-differential equations using Daftardar-Gejji and Jafari technique (DJM). *International Journal of Computer Mathematics*, *14*, 93–111.

Wazwaz, A. M. (2011). *Linear and nonlinear integral equations: Methods and applications*. Springer, Heidelberg, Dordrecht, London, New York.

Wazwaz, A. M. (2011). *Linear and nonlinear integral equations: Methods and applications*. Beijing: Higher Education Press; Berlin Heidelberg: Springer-Verlag.

Volterra, V. (1959). *Theory of functional and of integral and integro-differential equations*. New York: Dover Publications.