International Journal of Computational Methods
(2024) 2450056 (19 pages)

DOI: 10.1142/S0219876224500567

Numerical Solutions of Volterra Integral Equations with Subdivision-Based Collocation Approach

Zainab Iqbal* and Syeda Tehmina Ejaz†

Department of Mathematics, The Government Sadiq College Women University Bahawalpur, Bahawalpur 63100 Punjab, Pakistan
**zainabiqbal104@gmail.com*
†syedatehmina.ejaz@gscwu.edu.pk

Ali Akgül

Department of Mathematics, Faculty of Arts and Sciences Siirt University, 56100 Siirt, Turkey

Department of Computer Science and Mathematics Lebanese American University, P. O. Box 13-5053 Chouran, Beirut 1102 2801, Lebanon
aliakgul00727@gmail.com

Ureen Javaid

Department of Mathematics, The Government Sadiq College Women University Bahawalpur, Bahawalpur 63100 Punjab, Pakistan
ureenjavaid@gmail.com

Received 23 April 2024
Revised 6 July 2024
Accepted 6 September 2024
Published 4 November 2024

Integral equations play a vital role in the fields of applied and computational mathematics. This paper introduces a unique subdivision-based collocation technique for solving the initial value problems (IVPs) arising from Volterra integral equations (VIEs). A suitable subdivision scheme is employed to develop the numerical algorithm for solving the VIEs. Several numerical illustrations are tested and compared with the existing methods for assessing the efficiency of the proposed numerical algorithm and error estimation.

Keywords: Integral equation; Volterra integral equation; initial value problem; subdivision scheme; collocation algorithm.

†Corresponding author.

1. Introduction

The equations in which the unknown function appears under the integral symbol are known as integral equations. These equations have a wide range of applications in science and engineering. Integral equations are considered one of the most powerful tools in both pure and applied mathematics, as they are used to approximate many initial and boundary value problems [Usta (2021)].

The basic concepts regarding integral equations were studied by Rahman [2007]. He illustrated that on the basis of integral structures, initially, the integral equations were categorized into three forms namely Fredholm integral equations, Volterra integral equations (VIEs) and Fredholm–Volterra integral equations. du Bois-Reymond was the first one to use the word "integral equation" for any equation involving unknown function(s) under the sign of integration in 1888. Moreover, the Italian Mathematician Vito Volterra and the Swedish Mathematician Ivar Fredholm are considered the pioneers of the theory of integral equations.

The most general type of linear integral equation is given as follows [Wazwaz (2015)]:

$$Y(\psi)Z_E(\psi) = A(\psi) + \eta \int_{\mu}^{\psi} \delta(\psi, \phi) Z_E(\phi) d\phi, \tag{1}$$

where $Z_E(\psi)$ is an unknown function and μ is a real constant. The function $\delta(\psi, \phi)$ is called the kernel of integral equation defined in the rectangular domain $\mu \leq \psi \leq \nu$ and $\mu \leq \phi \leq \nu$ of the $\psi\phi$-plane. In Eq. (1), the upper limit of the integral is a variable represented by the parameter ψ. The functions $A(\psi)$, $Y(\psi)$ and $\delta(\psi, \phi)$ are known functions, while $Z_E(\psi)$ is an unknown function to be determined and η is a nonzero real or complex parameter. It is the most general type of Volterra integral equation which is also called the Volterra integral equation of the third kind. Furthermore, it can be classified into three forms. By setting $Y(\psi) = 0$ and $Y(\psi) = 1$ in Eq. (1), the Volterra integral equations of the first kind and second kind, respectively, are formed. Also by setting $Y(\psi) = 1$ and $A(\psi) = 0$ in Eq. (1), the homogeneous VIEs are obtained.

Volterra integral equations appear in various fields of physics and engineering, such as heat conduction in a rod, population dynamics, hereditary mechanics, electromagnetic wave propagation in dispersive media and viscoelastic materials. These examples illustrate the application of Volterra integral equations in modeling systems where the current state depends on the entire history of the system, which is a common scenario in many physical phenomena. For example, if we take $Y(\psi) = 1$ in (1), the obtained equation represents the mathematical model of heat conduction in a rod in which the term $Z_E(\psi)$ represents the temperature distribution at position ψ, $A(\psi)$ represents the initial temperature distribution and $\delta(\psi, \phi)$ represents the heat conduction property of the rod between the points ψ and ϕ, including the external heat source.

In the literature, several numerical approaches have been developed to solve these types of equations, such as the collocation method based on radial

basis functions, the Legendre collocation method and the Chebyshev polynomial approach. In addition, an approximating approach to solve VIEs of the second kind with a weakly singular kernel was introduced by Kong *et al.* [2023]. Quick and effective numerical approximations of partial integro-differential equations (PIDEs), which appear in financial problems and option pricing theory as well as in diverse scientific modeling, were studied by Bhowmik [2014a]. To strengthen the convergence requirements of iterative solvers for the complete discrete model, the author created preconditioners employing wavelet basis transformations and Fourier sine transformations. An effective numerical approach for linear mth-order boundary value problems was introduced by Bhowmik [2014b]. The higher-order boundary value problem was first converted to a first-order boundary value problem. Next, the boundary value problem solution was approximated as a weighted sum of polynomials using Chebyshev orthogonal polynomials. To create a system of equations that approximates the weights for the polynomials, the author collocated at Chebyshev clustered grid points. Furthermore, another study proposed and examined two different numerical approaches, namely the piecewise collocation finite-element and Galerkin finite-element approaches, for the numerical solution of nonlinear partial integro-differential problems that emerge during phase transition modeling. The study utilized certain common quadrature criteria to compute the solutions in both approaches [Bhowmik (2014c)].

In recent years, for the solutions of the linear and nonlinear integral equations many numerical techniques have been developed, including the basis function technique, Taylor's function approach, hat function methods and many other numerical techniques, as in many cases the analytic solutions of integral equations were not easy to obtain [Hesameddini and Shahbazi (2017)]. Likewise, in order to get the approximate solution of nonlinear Volterra integral equations of the first kind, a numerical approach based on Haar wavelets was presented by Singh and Kumar [2016]. In the same manner, approximation methods namely nonpolynomial-based spline function and quasi-linearization were introduced to solve the nonlinear VIE of the second kind numerically [Maleknejad *et al.* (2018)]. Similarly, an iterative technique was suggested to solve nonlinear Volterra equations based on piecewise smooth kernels [Sidorov *et al.* (2020)]. Furthermore, an iterative collocation-based approach was developed by using Lagrange polynomial to solve the nonlinear system of VIEs [Rouibah *et al.* (2022)].

Our novel proposed numerical algorithm, founded on the basis function and its derivatives of the interpolating subdivision scheme, introduces a unique foundation for handling the second-order spatial derivatives [Qu (1996)]. After developing these derivatives, Qu and Agarwal [1996, 1998] used these results to develop a collocation algorithm for solving second-order boundary value problems. Moreover, recent developments in this area of research, such as subdivision schemes based on various numerical algorithms to solve the nth-order linear and nonlinear ordinary differential equations, were introduced by Ejaz *et al.* [2021]. It is pertinent to mention that in this paper, we are interested to solve the VIEs numerically by using collocation

algorithm based on suitable subdivision schemes, as the subdivision schemes-based technique has not yet been employed earlier to obtain the numerical solution of VIEs in the literature. Moreover, it is evident from numerical and graphical findings that our technique is novel, robust and efficient when compared to the existing techniques presented by Zarnan [2016] and Nadir and Rahmoune [2018].

Subdivision schemes are used to generate smooth curves in geometric modeling. An appropriate nth-order continuous refinement scheme is formed by scheme (2) [Ejaz *et al.* (2021)],

$$\begin{cases} S_{2i}^{k+1} = S_i^k, \\ S_{2i+1}^{k+1} = \displaystyle\sum_{j=0}^{\lambda_1} \tau_{\lambda_1,j}(S_{i-j}^k + S_{i+j+1}^k), \end{cases} \tag{2}$$

where the coefficients in affine combination of scheme (2) are defined in (3),

$$\tau_{\lambda_1,j} = \frac{((2\lambda_1+1)!!)^2}{2(4\lambda^1)(2\lambda_1+1)!}\frac{(-1)^j}{(2j+1)}\binom{2\lambda_1+1}{\lambda_1-j}, \tag{3}$$

where $\binom{2\lambda_1+1}{\lambda_1-j}$ shows the binomial coefficient and λ_1 denotes the complexity of the scheme. By choosing appropriate values of the complexity parameter λ_1, various schemes can be constructed. If $\lambda_1 = 0, 1, 2, 3, \ldots$, then the complexities of the scheme will be $2, 4, 6, 8, \ldots$, respectively. To form a numerical algorithm for nth-order initial value problems (IVPs), an nth-order continuous scheme should be selected. Some important properties of the generalized binary interpolating subdivision scheme by Ejaz *et al.* [2021] are illustrated below:

- The basis function for scheme (2) is defined as follows:

$$\chi(i) = \begin{cases} 1 & \text{if } i = 0, \\ 0 & \text{if } i \neq 0. \end{cases}$$

- A local refinement matrix of invariant size $4\lambda_1 + 1$ is constructed from the coefficients of scheme (2).
- The eigenvalues ω_k and their right eigenvectors V_k corresponding to the subdivision matrix are defined as follows:

$$\omega_k = 2^{-k}, \quad 0 \leq k \leq 2\lambda_1 + 1,$$

$$V_k = (-2\lambda_1^k, (-2\lambda_1+1)^k, \ldots, (-1)^k, 0, 1, 2^k, \ldots, (2\lambda_1-1)^k, (2\lambda_1)^k)^T_{(4\lambda_1+1)\times 1}.$$

By taking the transpose of the local iteration matrix, the left eigenvectors L_k corresponding to the eigenvalues ω_k can be determined. The following relation must be satisfied by the eigenvectors:

$$(V_i)^T L_j = \delta_i \delta_j, \quad i, j = 1, 2, 3, \ldots, \lambda_1,$$

where

$$V_i^T L_j = 1 \quad \text{if } i = j,$$
$$V_i^T L_j = 0 \quad \text{if } i \neq j.$$

- The kth-order derivatives of the function $\chi(\psi)$ associated with the left eigenvector L_k are k times continuously differentiable over the interval $(-2\lambda_1 - 1, 2\lambda_1 + 1)$,

$$\chi^k(\psi) = 2^k(\text{sgn}(\psi))^k M_{|\psi|}^T L_k, \quad -2\lambda_1 \leq \psi \leq 2\lambda_1, \tag{4}$$

where

$$\text{sgn}(\psi) = \begin{cases} -1 & \text{if } \psi < 0, \\ 0 & \text{if } \psi = 0, \\ 1 & \text{if } \psi > 0 \end{cases}$$

and

$$M_\psi = (g_{(2k)\psi}, g_{(2k-1)\psi}, \ldots, g_{1\psi}, g_{0\psi}, g_{-1\psi}, g_{-2\psi}, \ldots, g_{(-2k+2)\psi},$$
$$g_{(-2k+1)\psi}, g_{(-2k)\psi}) \quad \text{for } 0 \leq \psi \leq 4\lambda_1 + 1.$$

- The smoothness order of the scheme selected should be same as that of IVP formed.
- Scheme (2) has $2\lambda_1 + 1$ degrees of generation and reproduction.
- $2\lambda_1 + 2$ is the approximation order of scheme (2).
- $(-2\lambda_1 - 1, 2\lambda_1 + 1)$ is the support of $\chi(\psi)$.

The structure of this paper is as follows: Section 2 is dedicated to the theoretical framework of our technique. Section 3 addresses the conversion of VIEs into IVPs. Section 4 discusses the formulation of the subdivision-based collocation algorithm. Furthermore, numerical examples along with results and discussions are presented in Sec. 5. Finally, Sec. 6 provides the concluding remarks.

2. Theoretical Framework

This section illustrates the step-wise theoretical framework of our technique to solve the VIEs numerically. In the first step, the VIEs are converted to nth-order IVPs. In the second step, a linear binary interpolating subdivision scheme is selected having same order of smoothness as that of IVP formed. The basis function of the scheme chosen must be nth-order continuously differentiable over the given interval range. Furthermore, the scheme chosen must attain the approximation order as per the support of the scheme. In the third step, an unstable/inconsistent system of algebraic equations is obtained by applying the basis function of the scheme selected. The inconsistent system of equations is made consistent by using any of the initial conditions or any extrapolation method. Then the stable/consistent system is solved by appropriate numerical method. Finally, the solution of the system of algebraic equations serves as the numerical solution of IVPs obtained by VIEs.

3. Step I: Formation of Initial Value Problems

This section presents the first step of our proposed algorithm which is about the transformation of VIEs into IVPs. There are mainly two approaches to convert the VIEs into IVPs, namely the resolvent kernel-based approach and Leibniz rule of differentiation.

First, we discuss the resolvent kernel-based approach to convert the VIEs into IVPs. For this, we consider a nonhomogeneous second-kind VIE [Rahman (2007)] as follows:

$$Z_E(\psi) = A(\psi) + \eta \int_a^{\psi} \delta(\psi, \phi) Z_E(\phi) d\phi. \tag{5}$$

Equation (5) has two possible forms of kernels [Rahman (2007)] as defined in (6) and (7),

$$\delta(\psi, \phi) = \xi_0(\psi) + \xi_1(\psi)(\psi - \phi) + \xi_2(\psi)\frac{(\psi - \phi)^2}{2!} + \cdots + \xi_n(\psi)\frac{(\psi - \phi)^{n-1}}{(n-1)!}, \tag{6}$$

$$\delta(\psi, \phi) = \zeta_0(\phi) + \zeta_1(\phi)(\phi - \psi) + \zeta_2(\phi)\frac{(\phi - \psi)^2}{2!} + \cdots + \zeta_{n-1}(\phi)\frac{(\phi - \psi)^{n-1}}{(n-1)!}. \tag{7}$$

In (6) and (7), the kernel $\delta(\psi, \phi)$ of Eq. (5) is the polynomial of degree $(n-1)$ in ψ and ϕ, respectively. The resolvent kernels for (5) corresponding to the kernels (6) and (7) are given in (8) and (9), respectively,

$$Q(\psi, \phi; \eta) = \frac{1}{\eta}\frac{d^n b(\psi, \phi; \eta)}{d\psi^n}, \tag{8}$$

$$Q(\psi, \phi; \eta) = -\frac{1}{\eta}\frac{d^n c(\psi, \phi; \eta)}{d\phi^n}, \tag{9}$$

where $b(\psi, \phi; \eta)$ and $c(\psi, \phi; \eta)$ are the solutions of the following differential equations, respectively:

$$\frac{d^n b}{d\psi^n} - \eta\left[\xi_0(\psi)\frac{d^{n-1}b}{d\psi^{n-1}} + \xi_1(\psi)\frac{d^{n-2}b}{d\psi^{2n-2}} + \cdots + \xi_{n-1}(\psi)b\right] = 0, \tag{10}$$

$$\frac{d^n c}{d\phi^n} + \eta\left[\zeta_0(\phi)\frac{d^{n-1}c}{d\phi^{n-1}} + \zeta_1(\phi)\frac{d^{n-2}c}{d\phi^{2n-2}} + \cdots + \zeta_{n-1}(\phi)c\right] = 0, \tag{11}$$

satisfying the following initial conditions, respectively:

$$\begin{cases} b = \dfrac{db}{d\psi} = \dfrac{d^2b}{d\psi^2} = \cdots = \dfrac{d^{n-2}b}{d\psi^{n-2}} = 0 & \text{when } \psi = \phi, \\ \dfrac{d^{n-1}b}{d\psi^{n-1}} = 1 & \text{when } \psi = \phi, \end{cases} \tag{12}$$

$$\begin{cases} c = \dfrac{dc}{d\phi} = \dfrac{d^2c}{d\phi^2} = \cdots = \dfrac{d^{n-2}c}{d\phi^{n-2}} = 0 & \text{when } \phi = \psi, \\ \dfrac{d^{n-1}c}{d\phi^{n-1}} = 1 & \text{when } \phi = \psi. \end{cases} \tag{13}$$

Thus, the solution of (5) is obtained by solving (10) with (12) or (11) with (13) depending on the type of kernel (8) or (9), respectively.

Second, we present a method to convert the integral equation (5) with the kernel defined in (6) or (7) into an IVP using the Leibniz rule of differentiation. For this, substituting Eq. (6) in (5), we get

$$Z_E(\psi) = A(\psi) + \eta \int_0^{\psi} \left[\xi_0 + \xi_1(\psi)(\psi - \phi) + \xi_2(\psi)\frac{(\psi-\phi)^2}{2!} + \cdots + \xi_{n-1}(\psi)\frac{(\psi-\phi)^{n-1}}{(n-1)!}\right] Z_E(\phi)d\phi. \tag{14}$$

Now, differentiating (14) with respect to ψ using the Leibniz rule until the integral term is removed, say up to the nth order, we get

$$\begin{cases} Z'_E(\psi) = A'(\psi) + \eta\xi_0(\psi)Z_E(\psi) + \eta\displaystyle\int_{\mu}^{\nu} \xi_1(\psi)Z_E(\phi)d\phi \\ \qquad\qquad + \cdots + \eta\displaystyle\int_{\mu}^{\nu} \xi_{n-1}(\psi)\frac{(\psi-\phi)^{n-2}}{(n-2)!}Z_E(\phi)d\phi, \\ Z''_E(\psi) = A''(\psi) + \eta\xi_0(\psi)Z'_E(\psi) + \eta\xi_1(\psi)Z_E(\psi) \\ \qquad\qquad + \cdots + \eta\displaystyle\int_{\mu}^{\nu} \xi_{n-1}(\psi)\frac{(\psi-\phi)^{n-3}}{(n-3)!}Z_E(\phi)d\phi, \\ \qquad\vdots \\ Z^n_E(\psi) = A^n(\psi) + \eta\xi_0(\psi)Z^{n-1}_E(\psi) + \eta\xi_1(\psi)Z^{n-2}_E(\psi) \\ \qquad\qquad + \cdots + \eta\xi_{n-1}(\psi)Z_E(\psi). \end{cases} \tag{15}$$

The initial conditions are obtained by substituting $\psi = \mu$ in (14) and (15), except for the nth-order derivative equation, and these are given in (16),

$$\begin{cases} Z_E(\mu) = A(\mu), \\ Z'_E(\mu) = A'(\mu) + \eta\xi_0(\mu)Z_E(\mu), \\ Z''_E(\mu) = A''(\mu) + \eta\xi_0(\mu)Z'_E(\mu) + \eta\xi_1(\mu)Z_E(\mu), \\ \qquad\vdots \\ Z^{n-1}_E(\mu) = A^{n-1}(\mu) + \eta\xi_0(\mu)Z^{n-2}_E(\mu) + \eta\xi_1(\mu)Z^{n-3}_E(\mu) \\ \qquad\qquad + \cdots + \eta\xi_n(\mu)Z_E(\mu). \end{cases} \tag{16}$$

Finally, the IVP (17) with the initial conditions given in (18) is obtained, respectively,

$$Z^n_E(\psi) - \eta\xi_0(\psi)Z^{n-1}_E(\psi) - \eta\xi_1(\psi)Z^{n-2}_E(\psi) - \cdots - \eta\xi_n(\psi)Z_E(\psi) = A^n(\psi), \tag{17}$$

$$\begin{cases} Z_E(\mu) = A(\mu), \\ Z'_E(\mu) = A'(\mu) + \eta\xi_0(\mu)Z_E(\mu), \\ Z''_E(\mu) = A''(\mu) + \eta\xi_0(\mu)Z'_E(\mu) + \eta\xi_1(\mu)Z_E(\mu), \\ \qquad \vdots \\ Z_E^{n-1}(\mu) = A^{n-1}(\mu) + \eta\xi_0(\mu)Z_E^{n-2}(\mu) + \eta\xi_1(\mu)Z_E^{n-3}(\mu) + \cdots + \eta\xi_n Z_E(\mu). \end{cases} \tag{18}$$

Thus, by solving Eq. (17) with (18), the required solution of VIE (5) is obtained. Similar expression can be obtained by substituting Eq. (7) into (5).

3.1. *Illustrative examples of IVPs*

This subsection highlights some illustrative examples regarding the conversion of VIEs into IVPs.

Example 3.1. Let us consider the following integral equation:

$$Z_E(\psi) = \psi + \int_0^{\psi} (\phi - \psi)Z(\psi)d\psi. \tag{19}$$

After applying one of the methods discussed previously to (19), we get the IVP given in (20),

$$Z''_E(\psi) + Z_E(\psi) = 0, \tag{20}$$

subject to the set of initial constraints (21), namely

$$\begin{cases} Z_E(0) = 0, \\ Z'_E(0) = 1, \end{cases} \tag{21}$$

which is the required second-order IVP.

Example 3.2. Let us consider the following integral equation:

$$Z_E(\psi) = 1 + 2\int_0^{\psi} (\psi - \phi)^3 Z_E(\phi)d\phi. \tag{22}$$

We have constructed the IVP given in (23) by applying one of the previously mentioned approaches (Leibniz rule or resolvent kernel approach) to (22), subject to the initial constraints provided in (24),

$$Z''''_E(\psi) - 12Z_E(\psi) = 0, \tag{23}$$

$$\begin{cases} Z_E(0) = 1, \\ Z'_E(0) = Z''_E(0) = Z'''_E(0) = 0. \end{cases} \tag{24}$$

Thus, the VIE (22) is converted into a fourth-order IVP.

4. Step II: Subdivision-Based Collocation Approach to Solve IVPs Generated by VIEs

This section presents the second step of our proposed numerical algorithm. Here, we introduce the generalized subdivision scheme-based collocation algorithm for solving the IVP defined in (17) with (18) obtained from the VIE (5) in detail. Let us consider the approximate solution of the general IVP defined in (17) with (18) for $\nu \leq \psi \leq \mu$, $N \geq 2\lambda_1$ as follows:

$$Z_E(\psi_j) = \sum_{i=-2\lambda_1}^{N+2\lambda_1} C_i \chi\left(\frac{\psi_j - \psi_i}{\hbar}\right) \quad \text{for } j = 0, 1, \ldots, N, \tag{25}$$

where $\psi_i = i\hbar$, $\psi_j = j\hbar$ and $\hbar = \frac{\nu-\mu}{N}$ is the step size. The pth-order derivatives of Eq. (25) are given as follows:

$$Z_E^p(\psi_j) = \frac{1}{\hbar^p} \sum_{i=-2\lambda_1}^{N+2\lambda_1} C_i \chi^p\left(\frac{\psi_j - \psi_i}{\hbar}\right) = \frac{1}{\hbar^p} \sum_{i=-2\lambda_1}^{N+2\lambda_1} C_i \chi^p(j-i), \quad p = 1, 2, \ldots, n. \tag{26}$$

Using (25) and (26) in (17), we have

$$\sum_{i=-2\lambda_1}^{N+2\lambda_1} C_i \eta \xi_0 \chi^n\left(\frac{\psi_j - \psi_i}{\hbar}\right) - \hbar^n \left[\frac{1}{\hbar^{n-1}} \sum_{i=-2\lambda_1}^{N+2\lambda_1} C_i \eta \xi_{n-1} \chi^{2n-1}\left(\frac{\psi_j - \psi_i}{\hbar}\right) - \cdots - \sum_{i=-2\lambda_1}^{N+2n_1} C_i \eta \xi_n \chi\left(\frac{\psi_j - \psi_i}{\hbar}\right)\right] = \hbar^n A(\psi_j), \quad j = 1, 2, \ldots, N.$$

The above equation can be simplified as

$$\sum_{i=-2\lambda_1}^{N+2\lambda_1} C_i \eta \xi_0 \chi^n(j-i) - \hbar^n \left[\frac{1}{\hbar^{n-1}} \sum_{i=-2\lambda_1}^{N+2\lambda_1} C_i \eta \xi_1 \chi^{n-1}(j-i) - \cdots - \sum_{i=-2\lambda_1}^{N+2\lambda_1} C_i \eta \xi_n \chi(j-i)\right] = \hbar^n A(\psi_j), \quad j = 1, 2, \ldots, N. \tag{27}$$

Using the notation $\chi_i^n = \chi^n(i)$ and $A(\psi_j) = A_j$ in (27), we get

$$\sum_{i=-2\lambda_1}^{N+2\lambda_1} C_i \eta [\xi_0 \chi_{j-i}^n - \hbar^n \{\xi_1 \chi_{j-i}^{n-1} - \cdots - \xi_n \chi_{j-i}\}] = \hbar^n A_j,$$

this can be expressed as

$$\sum_{i=-2\lambda_1}^{N+2\lambda_1} C_{i+j} B_{-i} = \hbar^n A_j, \quad j = 0, 1, 2, \ldots, N, \tag{28}$$

where

$$B_{-i} = \eta [\xi_0 \chi_{j-i}^n - \hbar^n \{\xi_1 \chi_{j-i}^{n-1} - \cdots - \xi_n \chi_{j-i}\}].$$

The matrix representation of (28) is

$$\mathbb{B}_E \mathbb{Z}_E = \mathbb{F}_E, \tag{29}$$

where the matrix $(\mathbb{B}_E)_{(N+1)\times(N+4\lambda_1+1)}$, column vectors $(\mathbb{Z}_E)_{N+4\lambda_1+1}$ and $(\mathbb{F}_E)_{N+1}$ are defined in (30) and (31), respectively,

$$\mathbb{B}_E = \begin{pmatrix} B_{-2\lambda_1} & B_{-2\lambda_1+1} & B_{-2\lambda_1+2} & \dots & B_{-1} & B_0 & B_1 & B_2 & \dots \\ 0 & B_{-2\lambda_1} & B_{-2\lambda_1+1} & \dots & B_{-2} & B_{-1} & B_0 & B_1 & \dots \\ 0 & 0 & B_{-2\lambda_1} & \dots & B_{-3} & B^n_{-2} & B_{-1} & B_2 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} B_{2\lambda_1} & 0 & 0 & 0 & \dots & 0 & 0 \\ B_{2\lambda_1-1} & B_{2\lambda_1} & 0 & 0 & \dots & 0 & 0 \\ B_{2\lambda_1-2} & B_{2\lambda_1-1} & B_{2\lambda_1} & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & B_{2\lambda_1-1} & B_{2\lambda_1} \end{pmatrix} \tag{30}$$

and

$$\begin{cases} \mathbb{Z}_E = (C_{-2\lambda_1}, C_{-2\lambda_1+1}, \dots, C_{-2\lambda_1+N-1}, C_{-2\lambda_1+N})^T, \\ \mathbb{F}_E = (\hbar^n A_0, \hbar^n A_1, \dots, \hbar^n A_N)^T. \end{cases} \tag{31}$$

System (29) is inconsistent as the number of equations and number of unknowns are not equal. In order to get a consistent system of (29), $4\lambda_1$ more equations are required. After adding the given initial conditions (18) to system (29), the order of the coefficient matrix increases but the system remains inconsistent. To attain a unique solution, $(4\lambda_1 - n)$ additional conditions are required.

If (17) is an even-order IVP, then $(4\lambda_1 - n)$ is divided at both ends of the domain as $\frac{4\lambda_1 - n}{2} = \alpha_1$ conditions are constructed at both the right and left ends of the domain of the above IVP. Similarly, if (17) is an odd-order IVP, then $(4\lambda_1 - n)$ is divided at both ends of the domain as $\frac{4\lambda_1 - n + 1}{2} = \alpha_2$ conditions are constructed at the left end and $\frac{4\lambda_1 - n - 1}{2} = \alpha_3$ conditions are constructed at the right end of system (29). These conditions are constructed in the next subsection.

4.1. *Approximation of derivative conditions and formation of extra conditions*

This subsection deals with the approximation of derivative conditions and construction of extra conditions for system (29) to be consistent.

4.1.1. *Approximation of the derivative conditions*

The derivative conditions (18) are approximated by adopting the following procedure. If $F(\psi)$ is a function, then for $\hbar > 0$ and integer $t > 0$ the mth derivative of $Z_E(\psi)$ can be approximated by the finite difference method as follows:

$$Z_E^m(\psi) = \frac{m!}{\hbar^m} \sum_{i=i_{\min}}^{i_{\max}} v_i F(\psi + i\hbar) + O(\hbar^t), \tag{32}$$

and the necessary condition for (32) to be satisfied is

$$\sum_{i=i_{\min}}^{i_{\max}} i^n v_i = \begin{cases} 0 & \text{for } 0 \le m + t - 1 \quad \text{and} \quad n \ne m, \\ 1 & \text{for } n = m, \end{cases} \tag{33}$$

for $i_{\min} = 0$ and $i_{\max} = m + t - 1$, the forward difference approximation has been used. The convolution mask is represented by the vector $V = (v_{i_{\min}}, \ldots, v_{i_{\max}})$. If we solve the system, then we get the convolution matrix V through which we can construct the approximation of the derivative conditions.

4.1.2. *Extra conditions*

Furthermore, the extra conditions are constructed through extrapolation method by selecting appropriate form of the polynomial. If $S_1(\psi)$ is the polynomial which interpolates the data (ψ_i, C_i) and $i \in [0, 2\lambda_1]$, then the values $C_{-(2\lambda_1-1)}$, $C_{-(2\lambda_1)}, \ldots, C_{-2}, C_{-1}$ for the left end can be computed. It is illustrated as

$$C_{-i} = S_1(-\psi_i),$$

for

$$i = \begin{cases} 1, 2, \ldots, \alpha_1 & \text{if (17) is an even-order IVP}, \\ 1, 2, \ldots, \alpha_2 & \text{if (17) is an odd-order IVP}, \end{cases}$$

where

$$S_1(\psi_i) = \sum_{j=1}^{2\lambda_1+2} \binom{2\lambda_1 + 2}{j} (-1)^{j+1} Z_E(\psi_{i-j});$$

now if we replace ψ_i by $-\psi_i$ for $i = 1, 2, \ldots, 2\lambda_1 + 1$, then

$$S_1(-\psi_i) = \sum_{j=1}^{2\lambda_1+2} \binom{2\lambda_1 + 2}{j} (-1)^{j+1} C_{-i+j},$$

so the left-end conditions are obtained as

$$\sum_{j=0}^{2\lambda_1+2} \binom{2\lambda_1 + 2}{j} (-1)^j C_{-i+j} = 0, \tag{34}$$

and similarly, for the right-end condition, we consider $C_i = S_1(\psi_i)$, where

$$i = \begin{cases} N+1, N+2, \ldots, \alpha_1 & \text{if (17) is an even-order IVP,} \\ N+1, N+2, \ldots, \alpha_3 & \text{if (17) is an odd-order IVP.} \end{cases}$$

Thus, the right-end conditions are constructed as

$$\sum_{j=0}^{2\lambda_1+2} \binom{2\lambda_1+2}{j} (-1)^j C_{-i+j} = 0. \tag{35}$$

Thus, by combining (34) and (35), the required extra conditions are obtained. Hence, the consistent system is obtained in the following form:

$$\mathbb{G}_E \mathbb{Z}_E = \mathbb{D}_E, \tag{36}$$

where

$$\mathbb{G}_E = ((\mathbb{B}_L)^T, (\mathbb{B}_E)^T, (\mathbb{B}_R)^T)^T. \tag{37}$$

In (36), the matrix $\mathbb{Z}_E$ is defined in (31), while the matrix $\mathbb{B}_E$ is given in (30). The same method presented in Ejaz *et al.* [2021] is adopted to construct the matrices $\mathbb{B}_L$ and $\mathbb{B}_R$ using (18), (34) and (35) and the order of these matrices ($\frac{4\lambda_1}{2} \times (N + 4\lambda_1 + 1)$). The matrix $\mathbb{D}_E$ is defined below:

$$\begin{aligned} \mathbb{D}_E = (0, \ldots, 0, 0, Z_E^{n-1}(\mu), Z_E^{n-2}(\mu), \ldots, Z'(\mu), Z(\mu), \mathbb{F}_E^T, Z(\nu), \\ Z'(\nu), \ldots, Z^{n-2}(\nu), Z^{n-1}(\nu), 0, 0, \ldots, 0)^T, \end{aligned} \tag{38}$$

where $\mathbb{F}_E$ is defined in (31). If the derivative constraints are given, then they are approximated by (32) and (33) before using them. This completes the construction of numerical algorithm.

Hence, by solving (37), we obtain the approximated solutions of (5).

4.2. *Error estimation*

In this subsection, we describe the error analysis by comparing the exact solutions and approximate solutions of our proposed algorithm. The absolute error (E), maximum relative error (E_{MR}), norm-2 (L_2) and maximum absolute error (E_{MA}) are computed as follows:

$$E = \|\text{Exact solution} - \text{Approximate solution}\|,$$

$$E_{\mathrm{MR}} = \left\| \frac{\text{Exact solution} - \text{Approximate solution}}{\text{Exact solution}} \right\|_\infty,$$

$$L_2 = \|\text{Exact solution} - \text{Approximate solution}\|_2,$$

$$E_{\mathrm{MA}} = \max \|\text{Exact solution} - \text{Approximate solution}\|_\infty.$$

5. Numerical Examples

In this section, we consider some particular problems in the form of IVPs generated by VIEs of the second kind in order to test the efficiency of our proposed algorithm.

Example 5.1 ([Nadir and Rahmoune (2018)]). For

$$Z(\psi) = \int_0^{\psi} -0.1Z(\psi)d\psi - \frac{1}{6}\psi^3 + \psi, \tag{39}$$

the initial value problem generated by Eq. (39) is

$$Z''(\psi) = -0.1Z'(\psi) + \psi^2, \quad \psi \in [0,1], \tag{40}$$

subject to the initial conditions

$$\begin{cases} Z(0) = 0, \\ Z'(0) = 1. \end{cases}$$

The exact solution is

$$Z(\psi) = 100\psi - 5\psi^2 + 990(e^{(-0.1)\psi} - 1).$$

Example 5.2 ([Zarnan (2016)]). For

$$Z(\psi) = \psi + \int_0^{\psi} (\phi - \psi)Z(\phi)d\phi, \tag{41}$$

the initial value problem generated by Eq. (41) is

$$Z''(\psi) + Z(\psi) = 0, \tag{42}$$

subject to the initial conditions

$$\begin{cases} Z(0) = 0, \\ Z'(0) = 1. \end{cases}$$

The exact solution of Eq. (41) is given as follows:

$$Z(\psi) = \sin(\psi).$$

5.1. *Results and discussion*

The numerical and graphical results of Examples 5.1 and 5.2 are presented in this subsection. By implementing the proposed method, we obtained the results numerically and graphically as presented in Tables 1–8 and Figs. 1 and 2 corresponding to Examples 5.1 and 5.2. The results of Example 5.1 are represented by Tables 1–3 and Fig. 1, whereas the results of Example 5.2 are represented by Tables 4–8 and Fig. 2.

The comparison between exact and approximate solutions of the proposed method is presented by Table 1, while Table 2 shows the comparison of absolute

Table 1. Comparison of the exact and approximate solutions of our method for Example 5.1.

ψ	Exact solution	Approximate solution
0	0	0
0.2	0.196686573687700	0.196686573297515
0.4	0.381544760799947	0.381544760524406
0.6	0.546888248406241	0.546888248208816
0.8	0.685182922769400	0.685182922626046
1	0.741873418393180	0.741873418393180

Table 2. Comparison of the absolute errors for Example 5.1.

ψ	E by our method	E by EM [Nadir and Rahmoune (2018)]	E by TSM [Nadir and Rahmoune (2018)]
0	0	0	0
0.2	3.90184412557559E−10	2.313426E−03	3.159513E−04
0.4	2.75541589545014E−10	8.505139E−03	5.405862E−04
0.6	1.97424632197851E−10	1.845966E−02	5.990007E−04
0.8	1.43353662274137E−10	3.206456E−02	4.193082E−04
1	0	4.921040E−02	6.745093E−05

Table 3. Different errors for Example 5.1 of the proposed method.

N	E_{MR}	L_2	E_{MA}
10	2.0637E−08	2.6894E−08	1.7366E−08
20	8.8470E−10	1.1463E−09	7.4445E−10
30	3.3992E−11	4.3738E−11	2.8603E−11
40	1.1832E−12	1.5136E−12	9.9566E−13
50	3.7644E−14	4.7921E−14	3.1676E−14

Table 4. Comparison of the exact and approximate solutions of our method for Example 5.2.

ψ	Exact solution	Approximate solution
0	0	0
0.1	0.0998334166468282	0.0998334097969941
0.2	0.1986693307950610	0.1986693249422950
0.3	0.2955202066613400	0.2955202016431860
0.4	0.3894183423086510	0.3894183379918320
0.5	0.4794255386042030	0.4794255348789130
0.6	0.5646424733950350	0.5646424701704330
0.7	0.6442176872376910	0.6442176844383310
0.8	0.7173560908995230	0.7173560884625110
0.9	0.7833269096274830	0.7833269075002000
1	0.8414709848078970	0.8414709848078970

Table 5. Comparison of the absolute errors for Example 5.2 for $N = 10$.

ψ	E by our method	E by the method of Zarnan [2016]
0	0	0
0.1	6.84983400556227E−09	1.67E−04
0.2	5.85276635489684E−09	3.31E−04
0.3	5.01815372588865E−09	4.90E−04
0.4	4.31681840273868E−09	6.42E−04
0.5	3.72529024295076E−09	7.84E−04
0.6	3.22460280699488E−09	9.14E−04
0.7	2.79935996605474E−09	1.03E−03
0.8	2.43701137048191E−09	1.13E−03
0.9	2.12728357151803E−09	1.21E−03
1	0	1.28E−03

Table 6. Comparison of the exact and approximate solutions of our method for Example 5.2.

ψ	Exact solution	Approximate solution
0	0	0
0.05	0.0499791692706783	0.0499791624208443
0.1	0.0998334166468282	0.0998334097969941
0.15	0.1494381324735990	0.1494381274554460
0.2	0.1986693307950610	0.1986693249422950
0.25	0.2474039592545230	0.2474039555292330
0.3	0.2955202066613400	0.2955202016431860
0.35	0.3428978074554510	0.3428978046560910
0.4	0.3894183423086510	0.3894183379918320
0.45	0.4349655341112300	0.4349655319839470
0.5	0.4794255386042030	0.4794255348789130
0.55	0.5226872289306590	0.5226872272972600
0.6	0.5646424733950350	0.5646424701704330
0.65	0.6051864057360400	0.6051864044697510
0.7	0.6442176872376910	0.6442176844383310
0.75	0.6816387600233340	0.6816387590328380
0.8	0.7173560908995230	0.7173560884625110
0.85	0.7512804051402930	0.7512804043590430
0.9	0.7833269096274830	0.7833269075002000
0.95	0.8134155047893740	0.8134155041683520
1	0.8414709848078970	0.8414709848078970

errors of the present method with the errors of the Euler method (EM) and Taylor series method (TSM) presented in Nadir and Rahmoune [2018] for Example 5.1. Additionally, Table 3 shows the different errors for Example 5.1. The error estimation shows that our method is more accurate and convergent than the methods presented in Nadir and Rahmoune [2018]. Moreover, Fig. 1 shows the graphical display of exact and approximate solutions corresponding to Example 5.1.

Table 7. Comparison of the absolute errors for Example 5.2 for $N = 20$.

ψ	E by our method	E by the method of Zarnan [2016]
0	0	0
0.05	6.84983401250117E−09	2.08E−05
0.1	5.85276634101906E−09	4.16E−05
0.15	5.01815372588865E−09	6.22E−05
0.2	4.3168183749831E−09	8.25E−05
0.25	3.72529029846191E−09	1.03E−04
0.3	3.22460275148373E−09	1.22E−04
0.35	2.79936002156589E−09	1.41E−04
0.4	2.43701131497076E−09	1.60E−04
0.45	2.12728351600688E−09	1.78E−04
0.5	1.8617342667504E−09	1.96E−04
0.55	1.63339963776821E−09	2.12E−04
0.6	1.43651435191572E−09	2.28E−04
0.65	1.26628829644204E−09	2.43E−04
0.7	1.11872877628372E−09	2.57E−04
0.75	9.90496240582672E−10	2.70E−04
0.8	8.78789152558568E−10	2.82E−04
0.85	7.81249953618612E−10	2.93E−04
0.9	6.95888902058073E−10	3.03E−04
0.95	6.21022011593197E−10	3.12E−04
1	0	3.19E−04

Table 8. Different errors for Example 5.2 of the proposed method.

N	E_{MR}	L_2	E_{MA}
10	7.1141E−10	1.082E−09	5.6133E−10
20	4.9633E−11	6.8516E−11	3.1962E−11
30	3.4627E−12	4.5773E−12	2.7322E−12
40	2.4195E−13	3.0678E−13	1.9062E−13
50	1.6855E−14	2.0662E−14	1.3299E−14

The comparison of exact and approximate solutions of our method is shown by Tables 4 and 6 for $N = 10$ and $N = 20$, respectively, for Example 5.2. Similarly, the comparison of absolute errors of our method with the method presented in Zarnan [2016] has been showcased by Tables 5 and 7 for $N = 10$ and $N = 20$, respectively. In contrast, the graphical representation of exact and approximate solutions is displayed by Fig. 2 for Example 5.2. In addition, Table 8 represents the different errors for Example 5.2.

In all the tables and figures, the parameter ψ shows some nodal points of the interval range. From all the numerical and graphical findings, it is evident that our technique is more efficient and convergent than the techniques presented in Nadir and Rahmoune [2018] and Zarnan [2016].

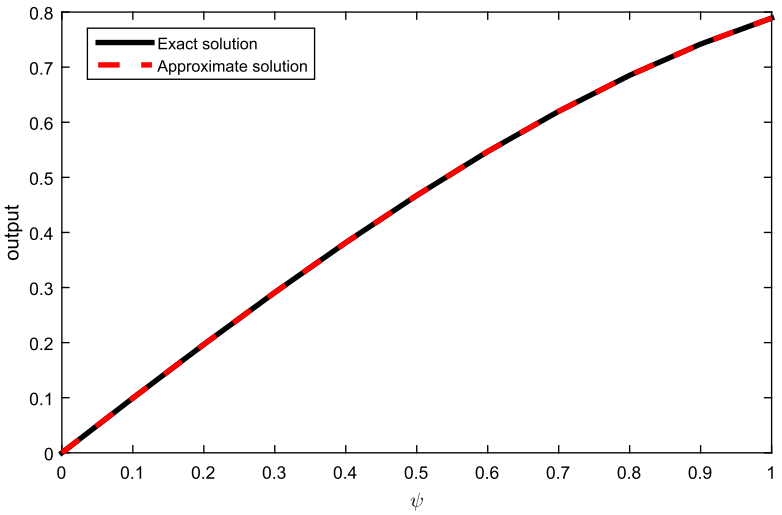

Fig. 1. Graphical representation of Example 5.1.

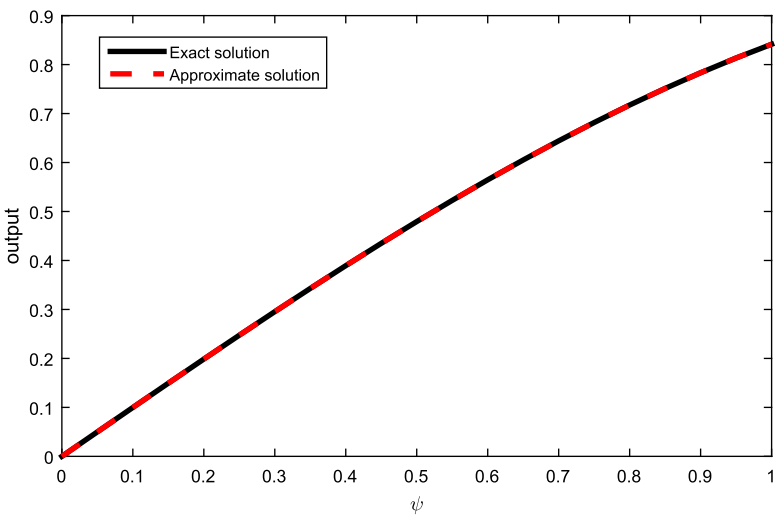

Fig. 2. Graphical representation of Example 5.2.

6. Concluding Remarks

In this paper, we present a numerical algorithm for the approximate solution of nth-order initial IVPs generated by VIEs. A suitable subdivision scheme has been used to develop the algorithm. In the light of numerical and graphical results, it is evident that for large values of N, the accuracy of our algorithm can be further

increased. In addition, the results of numerical illustrations reflect that our proposed algorithm is more efficient as compared to the existing methods presented in Nadir and Rahmoune [2018] and Zarnan [2016].

ORCID

Zainab Iqbal https://orcid.org/0009-0001-1287-7251
Syeda Tehmina Ejaz https://orcid.org/0000-0001-7405-1757
Ali Akgül https://orcid.org/0000-0001-9832-1424
Ureen Javaid https://orcid.org/0009-0007-5315-7051

References

Bhowmik, S. K. [2014a] "Fast and efficient numerical methods for an extended Black-Scholes model," *Comput. Math. Appl.* **67**(3), 636–654.
Bhowmik, S. K. [2014b] "Tchebychev polynomial approximations for m^{th} order boundary value problems," preprint, arXiv:1404.5032 [math.NA].
Bhowmik, S. K. [2014c] "Piecewise polynomial approximation of a nonlocal phase transitions model," *J. Math. Anal. Appl.* **420**(2), 1069–1094.
Ejaz, S. T. *et al.* [2021] "The refinement-schemes-based unified algorithms for certain nth order linear and nonlinear differential equations with a set of constraints," *Adv. Differ. Equ.* **2021**(1), 121.
Hesameddini, E. and Shahbazi, M. [2017] "Solving system of Volterra–Fredholm integral equations with Bernstein polynomials and hybrid Bernstein Block-Pulse functions," *J. Comput. Appl. Math.* **315**, 182–194.
Kong, D., Xiang, S. and Wu, H. [2023] "An efficient numerical method for Volterra integral equation of the second kind with a weakly singular kernel," *J. Comput. Appl. Math.* **427**, 115101.
Maleknejad, K., Rashidinia, J. and Jalilian, H. [2018] "Non-polynomial spline functions and quasi-linearization to approximate nonlinear Volterra integral equation," *Filomat* **32**(11), 3947–3956.
Nadir, M. R. and Rahmoune, A. [2018] "Initial value problems between Taylor and Volterra integral equations," *MATLAB J.* **1**(1), 14–25.
Qu, R. [1996] "A new approach to numerical differentiation and integration," *Math. Comput. Model.* **24**, 55–68.
Qu, R. and Agarwal, R. P. [1996] "Solving two point boundary value problems by interpolatory subdivision algorithms," *Int. J. Comput. Math.* **60**, 279–294.
Qu, R. and Agarwal, R. P. [1998] "A subdivision approach to the construction of approximate solutions of boundary-value problems with deviating arguments," *J. Comput. Math. Appl.* **35**(11), 121–135.
Rahman, M. [2007] *Integral Equations and Their Applications* (WIT Press, Southampton and Boston).
Rouibah, K. *et al.* [2022] "Iterative continuous collocation method for solving nonlinear Volterra integral equations," *Kragujev. J. Math.* **46**(4), 635–648.
Sidorov, D. *et al.* [2020] "Nonlinear systems of Volterra equations with piecewise smooth kernels: Numerical solution and application for power systems operation," *Mathematics* **8**(8), 1257.
Singh, I. and Kumar, S. [2016] "Haar wavelet method for some nonlinear Volterra integral equations of the first kind," *J. Comput. Appl. Math.* **292**, 541–552.

Usta, F. [2021] "Numerical analysis of fractional Volterra integral equations via Bernstein approximation method," *J. Comput. Appl. Math.* **384**, 113198.
Wazwaz, A. M. [2015] *A First Course in Integral Equations* (World Scientific, Singapore).
Zarnan, J. A. [2016] "Numerical solution of Volterra integral equations of second kind," *Int. J. Comput. Sci. Mob. Comput.* **5**(7), 509–517.