



An interpolation-based method for solving Volterra integral equations

Nasibeh Karamollahi¹ · Mohammad Heydari¹ · Ghasem Barid Loghmani¹

Received: 26 September 2020 / Revised: 20 November 2020 / Accepted: 27 March 2021 /

Published online: 19 April 2021

© Korean Society for Informatics and Computational Applied Mathematics 2021

Abstract

In this study, the second kind Volterra integral equations (VIEs) are considered. An algorithm based on the two-point Taylor formula as a special case of the Hermite interpolation is proposed to approximate the solution of such problems. The method can be applied for solving both the linear and nonlinear VIEs and systems of nonlinear VIEs. The convergence analysis and the error estimate of the method are described. A multistep form of the algorithm which is particularly beneficial in large intervals is also presented. The main advantage of the proposed algorithm is that it gives high accurate results in acceptable computational times. In order to indicate the validity of the method, it is employed for solving several illustrative examples. The efficiency of the method is confirmed through our study respecting the absolute errors and CPU times.

Keywords Volterra integral equations · Two-point Taylor formula · Multistep method · Convergence analysis

Mathematics Subject Classification 45D05 · 45G15 · 65D05 · 65R20

✉ Ghasem Barid Loghmani
loghmani@yazd.ac.ir

Nasibeh Karamollahi
karamollahi.n@gmail.com

Mohammad Heydari
m.heydari@yazd.ac.ir; m.heydari85@gmail.com

¹ Department of Mathematics, Yazd University, Yazd, Iran

1 Introduction

Consider the following VIE of the second kind

$$u(x) = f(x) + \lambda \int_a^x k(x, t, u(t))dt, \quad x \in I := [a, b], \tag{1.1}$$

where λ is a non-zero constant, the right hand side function f , and the kernel k are several times continuously differentiable with respect to x on the interval I , and u is the unknown solution of the problem.

VIEs of type (1.1) frequently appear in the diffusion problems [1], heat conduction [2], electroelastic [3], electromagnetics [4], and many other areas of science, engineering, economics, medicine, etc [5–10]. Consider, for instance, the problem of scattering of a electromagnetic wave with a kerr type nonlinear layer, which is an important problem in the nonlinear optics. Suppose that a plane wave of frequency ω_0 encounters a dielectric film of thickness d_0 at angle φ . The film is between two semi-infinite media and all the layers are assumed to be non-magnetic and homogenous. Considering a kerr-like nonlinearity inside the film, the permittivity function can be formulated as [8]

$$\varepsilon(y) = \begin{cases} \varepsilon_1, & y > d, \\ \varepsilon_2 = \varepsilon_2^0 + \tilde{\varepsilon}_2(y) + a_0 E^2(y), & 0 < y < d, \\ \varepsilon_3, & y < 0, \end{cases}$$

where $a_0, \varepsilon_1, \varepsilon_2^0, \varepsilon_3$ are real constants.

Utilizing the Maxwell equations, the problem can be modelled by the following nonlinear Helmholtz equations for each of the layers ($j = 1, 2, 3$) [8,11]

$$\frac{\partial^2 \tilde{E}_j(x, y)}{\partial x^2} + \frac{\partial^2 \tilde{E}_j(x, y)}{\partial y^2} + k_0^2 \varepsilon_j \tilde{E}_j(x, y) = 0, \quad j = 1, 2, 3,$$

where $k_0^2 = \omega_0^2/c^2$ and $\tilde{E}_j(x, y)$ is the time-independent part of the fields.

In order to investigate the field intensity inside the film, it is helpful to reduce the Helmholtz equation for the film ($j = 2$) into a nonlinear VIE as follows [8]:

$$I(y) = f(y) + \int_0^y k(y, t, I(t)) I(t)dt,$$

where

$$\begin{aligned} f(y) &= A \cos(2\kappa y) + B \sin(2\kappa y) + \frac{c_2}{2\kappa^2} \sin^2(\kappa y), \\ k(y, t, I(t)) &= -\frac{\sin(2\kappa(y-t))}{\kappa} \left(8\pi^2 \tilde{\varepsilon}_2(t) + 6\pi^2 a_0 I(t) \right) \\ &\quad - 2\pi^2 \frac{\cos(2\kappa(y-t)) - 1}{\kappa^2} \frac{d\tilde{\varepsilon}_2(t)}{dt}, \end{aligned}$$

$$\kappa^2 = 4\pi^2 \varepsilon_2^0 - p^2, \quad p = \sqrt{\varepsilon_1} k_0 \sin(\varphi).$$

As another important problem, we present here the susceptible–infective–susceptible (SIS) epidemic model which can be formulated as a VIE. This formulation can be advantageous to analyse the reproduction number of the disease and the methods for disease control. Let $I(t)$ and $S(t) = 1 - I(t)$ be the fraction of the infective and susceptible individuals, respectively. Suppose that $\lambda(I)$ is the contact rate, which represents the number of encounters per unit time between an infective person and another person in the population [12]. All the individuals that are not infective at time t belong to the susceptible population at this time. The rate $\lambda(I(t))I(t)S(t)$ shows the probability for a susceptible person to become infective. Let the fraction of the individuals that were infective from time zero to time t and survive to this time be denoted by $P(t)$, and let b_0 denote the death rate which is independent of infection. The mean time that a person remains infective can be computed as [13]

$$\tau = \int_0^\infty P(u)e^{-b_0 u} du.$$

Under the above assumptions, the fraction of infective individuals can be formulated by the following VIE [13]

$$I(t) = I_0(t) + \int_0^t \lambda(I(u)) I(u) (1 - I(u)) P(t - u) e^{-b_0(t-u)} du,$$

where $I_0(t)$ is the fraction of individuals that were infective from time zero to time t .

Since most of such problems can not be solved analytically, presenting and improving numerical techniques for solving VIEs have been an important branch of the numerical analysis in the last decades. Several researchers have attempted to present some efficient schemes to approximate the solution of VIEs. Conte and Paternoster in [14] proposed the multistep collocation methods for solving VIEs and generated stable solutions with low computational costs. Cherruault and Zitoun in [15] presented a simple and accurate method based on the Taylor expansion and the Faà di Bruno's formula to approximate the solution of VIEs of the second and third kind. Maleknejad et al. [16] employed the fixed point method (FPM) with the cubic B-spline scaling functions as basis functions to solve nonlinear VIEs. Also, Maleknejad and Najafi [17] combined the idea of quasilinearization with the Lagrange collocation method (LCQM) in the piecewise continuous space to solve nonlinear VIEs. In their presented method, nonlinear VIEs are solved indirectly with solving the related sequences of linear VIEs. The quasilinearization technique was employed later with non-polynomial spline functions (NPSQM) in [18] and gave more accurate results in comparison to the LCQM. Mirzaee and Hoseini utilized a combination of Bernstein polynomials and block-pulse functions to present a numerical method for solving a class of VIEs of the second kind [19]. Also, Mirzaee and Hadadiyan in [20] used the modified block-pulse functions method to approximate the solution of VIEs. The Nyström interpolation formula was utilized to solve VIE (1.1) by Baratella [1]. The Adomian decomposition method [21], homotopy analysis method [22], Galerkin method [23–26], radial basis

functions method [27–31], Bernoulli collocation method [32], Fibonacci collocation method [33], Euler matrix method [34], two-dimensional delta functions method [35], least squares method [36,37], trapezoidal quadrature iterative method [38], implicit Runge–Kutta method [39], reproducing kernel method [40], successive approximations method [41–43], modified Newton method [44], hat functions method [45–49], and Chebyshev cardinal functions method [50] can also be addressed as some effective methods which have been applied for solving VIEs in recent years. We refer the interested reader to [51–57] for more details about VIEs and their applications.

The main aim of this article is to present a simple and fast numerical method to solve linear and nonlinear VIEs. The proposed scheme is based upon a special type of Hermite interpolation method. In this interpolation problem, the needed data to construct the polynomial interpolant of an arbitrary function over each interval are the values of the function and its derivatives up to an adequate order at the endpoints of the interval. This interpolation problem is called the two-point Hermite interpolant or two-point Taylor formula (TTF). Phillips in [58] obtained an explicit form of the two-point Hermite interpolant over $[0, 1]$. This kind of interpolation was applied later to solve ordinary differential equations [59–61]. However, according to the authors knowledge, the utility of this method for solving Volterra integral equations has not been investigated before. The TTF uses a set of $2n$ data to construct a $(2n - 1)$ th degree polynomial as an approximate solution of a VIE. Since the structure of the problem provides us the half of the needed data, the solution of the VIE reduces to the solution of a system of n algebraic equations. The constructed system can be solved by using a suitable iterative method. Furthermore, we present the multistep TTF which is a reliable method to solve VIEs of type (1.1), especially in large domains, where the TTF may take difficulty to obtain accurate results. The main feature of the multistep TTF is that it uses in each step the data of the previous step. Therefore, in each step, only the half of the needed data remain unknown. Also, this property keeps the continuity of the solution. Since in this method the large intervals are divided into small subintervals, the multistep TTF can provide high accurate approximations without the need to increase the number of bases. Although many of the methods in the literature encounter difficulties for solving VIEs over large intervals, the multistep TTF gives acceptable results in such intervals. The error bound and the convergence analysis of the proposed method are studied. The capability of the method is inspected by applying it to solve different examples of VIEs of form (1.1). All the obtained results verify the analytical studies.

The next sections of this article are organized as follows. In Sect. 2, the TTF and its multistep form for solving VIEs of type (1.1) are explained. The convergence analysis of the proposed TTF is discussed in Sect. 3. The numerical results of applying the method for solving several VIEs are reported in Sect. 4. Finally, we finish this article with the conclusion of this study.

2 The TTF for solving VIEs

2.1 Construction of the TTF

Suppose that the values of a function $f \in C^{2n} [0, 1]$, and its derivatives up to $(n - 1)$ th order are given at $x = 0$ and $x = 1$. The TTF generates a polynomial $P \in \mathbb{P}_{2n-1}$ which satisfies the following given data:

$$\begin{cases} P^{(q)}(0) = f^{(q)}(0), \\ P^{(q)}(1) = f^{(q)}(1), \end{cases} \quad q = 0, 1, 2, \dots, n - 1. \tag{2.1}$$

An explicit form of $P(x)$ based on interpolation conditions (2.1) is illustrated in the following theorem.

Theorem 2.1 [58] *Suppose that $f \in C^{2n} [0, 1]$, then f can be approximated by using the following polynomial interpolant*

$$P_{2n-1}(f; x) = \sum_{j=0}^{n-1} \sum_{s=0}^1 (-1)^{js} \phi_{n,j}(s + (-1)^s x) f^{(j)}(s), \tag{2.2}$$

where $\phi_{n,j}(x)$, $j = 0, \dots, n - 1$ are the nonnegative basis functions on the interval $[0, 1]$ and defined as

$$\phi_{n,j}(x) = \frac{x^j}{j!} (1 - x)^n \sum_{k=0}^{n-j-1} \binom{n+k-1}{k} x^k.$$

Furthermore, there exists a number $\xi_x \in (0, 1)$ such that

$$R_n(f; x) := f(x) - P_{2n-1}(f; x) = \frac{f^{(2n)}(\xi_x)}{(2n)!} x^n (1 - x)^n.$$

In order to obtain the TTF over a general interval $[a, b]$, and approximate an arbitrary function $f \in C^{2n} [a, b]$, we can follow [58] and define the linear mapping $\psi := [a, b] \rightarrow [0, 1]$ as

$$\psi(x) = \frac{x - a}{b - a}.$$

Now, the basis functions $\tilde{\phi}_{n,j}$, $j = 0 \dots, n - 1$, can be constructed such that

$$\frac{d^q}{dx^q} \tilde{\phi}_{n,j}(x) = 0, \quad \text{at } x = b, \quad q = 0, 1, 2, \dots, n - 1,$$

and

$$\frac{d^q}{dx^q} \tilde{\phi}_{n,j}(x) = \delta_{jq}, \quad \text{at } x = a, \quad q = 0, 1, 2, \dots, n - 1,$$

where δ_{jq} is the Kronecker delta function. Therefore, we can state the following corollary.

Corollary 2.1 *Let $f \in C^{2n} [a, b]$, then*

$$f(x) = P_{2n-1}(f; x) + R_n(f; x),$$

where

$$P_{2n-1}(f; x) = \sum_{j=0}^{n-1} \sum_{s=0}^1 (-1)^{js} \tilde{\phi}_{n,j} \left((a+b)s + (-1)^s x \right) f^{(j)} \left((b-a)s + a \right), \tag{2.3}$$

$$\tilde{\phi}_{n,j}(x) = (b-a)^j (\phi_{n,j} \circ \psi)(x), \quad j = 0, \dots, n-1,$$

$$R_n(f; x) = \frac{f^{(2n)}(\xi_x)}{(2n)!} (x-a)^n (b-x)^n, \quad \xi_x \in (a, b). \tag{2.4}$$

Polynomial (2.3) satisfies the following interpolation conditions

$$\begin{cases} P_{2n-1}^{(q)}(f; a) = f^{(q)}(a), \\ P_{2n-1}^{(q)}(f; b) = f^{(q)}(b), \end{cases} \quad q = 0, 1, 2, \dots, n-1,$$

and gives an approximation to f for every point $x \in [a, b]$.

2.2 Implementation of the TTF for VIEs

Now, suppose that $u \in C^{2n} [a, b]$ is the exact solution of VIE (1.1). According to the description in the previous subsection, the TTF (2.3) can provide a $(2n - 1)$ th degree approximation to u as

$$u(x) \simeq P_{2n-1}(u; x) = \sum_{j=0}^{n-1} \sum_{s=0}^1 (-1)^{js} \tilde{\phi}_{n,j} \left((a+b)s + (-1)^s x \right) u^{(j)} \left((b-a)s + a \right). \tag{2.5}$$

The only needed data to construct (2.5) are $u^{(j)}(a)$ and $u^{(j)}(b)$ for $j = 0, 1, \dots, n-1$. In order to obtain them, we follow the idea of [15] and define the nonlinear Volterra integral operator $\mathcal{K} : C^{n-1}[a, b] \rightarrow C^{n-1}[a, b]$ as follows:

$$(\mathcal{K}u)(x) = \int_a^x k(x, t, u(t))dt.$$

Now, by j times differentiating both sides of VIE (1.1) with respect to x , we can deduce

$$u^{(j)}(x) = f^{(j)}(x) + \lambda(\mathcal{K}u)^{(j)}(x), \quad j = 0, 1, 2, \dots, n-1, \tag{2.6}$$

where

$$(\mathcal{K}u)^{(j)}(x) = \sum_{i=0}^{j-1} (\mathcal{D}_{j-i-1}u)^{(i)}(x) + \int_a^x \frac{\partial^j}{\partial x^j} k(x, t, u(t)) dt,$$

and

$$(\mathcal{D}_j u)(x) = \left. \frac{\partial^j}{\partial x^j} k(x, t, u(t)) \right|_{t=x}, \quad j = 0, 1, \dots, n - 1.$$

The values of $u^{(j)}(a)$, $j = 0, 1, \dots, n - 1$ can be calculated directly from (2.6) as

$$\begin{cases} u(a) = f(a), \\ u^{(j)}(a) = f^{(j)}(a) + \lambda \sum_{i=0}^{j-1} (\mathcal{D}_{j-i-1}u)^{(i)}(a), \quad j = 1, 2, \dots, n - 1. \end{cases} \quad (2.7)$$

Therefore, polynomial interpolant (2.5) includes only n unknowns $u^{(j)}(b)$, $j = 0, 1, \dots, n - 1$. Note that by substituting polynomial approximation (2.5) into (2.6), we get

$$u^{(j)}(x) \simeq f^{(j)}(x) + \lambda (\mathcal{K}P_{2n-1})^{(j)}(x), \quad j = 0, 1, 2, \dots, n - 1. \quad (2.8)$$

If $\tilde{u}^{(j)}(b)$ is the estimation of the exact value $u^{(j)}(b)$, an approximation to $P_{2n-1}(u; x)$ can be determined as

$$\tilde{P}_{2n-1}(u; x) = \sum_{j=0}^{n-1} \sum_{s=0}^1 (-1)^{js} \tilde{\phi}_{n,j} \left((a+b)s + (-1)^s x \right) \tilde{u}^{(j)} \left((b-a)s + a \right), \quad (2.9)$$

where

$$\begin{cases} \tilde{u}^{(j)}(a) = u^{(j)}(a), \\ \tilde{u}^{(j)}(b) \simeq u^{(j)}(b), \end{cases} \quad j = 0, 1, \dots, n - 1. \quad (2.10)$$

Indeed, the approximate values $\tilde{u}^{(j)}(b)$, $j = 0, 1, \dots, n - 1$ can be obtained by solving the following system of n equations

$$\tilde{u}^{(j)}(b) = f^{(j)}(b) + \lambda (\mathcal{K}\tilde{P}_{2n-1})^{(j)}(b), \quad j = 0, 1, 2, \dots, n - 1. \quad (2.11)$$

The obtained approximations from (2.11) can then be utilized to calculate polynomial (2.9) as an approximation to the exact solution of VIE (1.1).

Remark 2.1 The integrals which appeared in (2.11) can be accurately estimated by using the Gauss–Legendre quadrature formula as

$$\int_a^b \frac{\partial^j}{\partial x^j} k(x, t, \tilde{P}_{2n-1}(u; t)) \Big|_{x=b} dt \simeq \frac{b-a}{2} \sum_{j=0}^m w_j \frac{\partial^j}{\partial x^j} k(x, \tau_j, \tilde{P}_{2n-1}(u; \tau_j)) \Big|_{x=b}, \tag{2.12}$$

where

$$\begin{cases} \tau_j = \frac{b-a}{2} t_j + \frac{b+a}{2}, \\ w_j = \frac{2}{(1-t_j^2)(L'_{m+1}(t_j))^2}, \end{cases} \quad j = 0, 1, \dots, m,$$

and $\{t_j\}_{j=0}^m$ are the zeros of the Legendre polynomial $L_{m+1}(t)$.

2.3 The multistep TTF for VIEs

The presented TTF method only uses the data at the endpoints of the domain. This may increase the error of the approximation over large intervals. Therefore, in what follows, we explain the multistep form of the TTF for solving VIEs of the second kind. Let us discretize the interval I by using the uniform mesh $I_h := \{x_i := a + ih, i = 0, 1, \dots, N\}$, where $h = \frac{b-a}{N}$ is the mesh size and N is a positive integer. Consider again the nonlinear VIE (1.1) with the exact solution $u(x)$. Let $u_r, r = 1, 2, \dots, N$, be the exact solution of problem (1.1) over the interval $\sigma_r := [x_{r-1}, x_r]$, then this problem can be divided into a set of N problems as follows:

$$\begin{aligned} u_r(x) = & f(x) + \lambda \sum_{i=1}^{r-1} \int_{x_{i-1}}^{x_i} k(x, t, u_i(t)) dt \\ & + \lambda \int_{x_{r-1}}^x k(x, t, u_r(t)) dt, \quad r = 1, 2, \dots, N. \end{aligned} \tag{2.13}$$

Now, in each step, we can estimate the solution by using the TTF as

$$\begin{aligned} u_r(x) \simeq & P_{2n-1}^{[r]}(u; x) = \sum_{j=0}^{n-1} \\ & \sum_{s=0}^1 (-1)^{js} \tilde{\phi}_{n,j}^{[r]} \left((x_{r-1} + x_r)s + (-1)^s x \right) u^{(j)}(hs + x_{r-1}), \quad x \in \sigma_r, \end{aligned}$$

where

$$\begin{aligned} \psi^{[r]}(x) &= \frac{1}{h}(x - x_{r-1}), \quad r = 1, 2, \dots, N, \\ \tilde{\phi}_{n,j}^{[r]}(x) &= h^j (\phi_{n,j} \circ \psi^{[r]})(x), \quad j = 0 \dots, n - 1, \quad r = 1, 2, \dots, N. \end{aligned}$$

Differentiating both sides of (2.13) with respect to x leads to

$$u_r^{(j)}(x) = f^{(j)}(x) + \lambda \sum_{i=1}^{r-1} \int_{x_{i-1}}^{x_i} \frac{\partial^j}{\partial x^j} k(x, t, u_i(t)) dt + \lambda (\mathcal{K}_r u)^{(j)}(x), \tag{2.14}$$

where

$$(\mathcal{K}_r u)^{(j)}(x) = \sum_{i=0}^{j-1} (\mathcal{D}_{j-i-1} u_r)^{(i)}(x) + \int_{x_{r-1}}^x \frac{\partial^j}{\partial x^j} k(x, t, u_r(t)) dt.$$

In the first step, we require $u_1^{(j)}(x_0)$ and $u_1^{(j)}(x_1)$ for $j = 0, 1, \dots, n - 1$, to obtain the TTF on σ_1 . Similar to (2.7), the values of $u_1^{(j)}(x_0)$, $j = 0, 1, \dots, n - 1$, can be exactly calculated from (2.14). Thus, the only unknowns in this step are $u_1^{(j)}(x_1)$, $j = 0, 1, \dots, n - 1$. Furthermore, it can be concluded from (2.13) that

$$u_1^{(j)}(x_1) = u_2^{(j)}(x_1), \quad j = 0, 1, \dots, n - 1.$$

Therefore, if we compute the values of $u_1^{(j)}(x_1)$, $j = 0, 1, \dots, n - 1$ in the first step, they can be utilized again in the second step to obtain the TTF on σ_2 . Since generally we have

$$u_r^{(j)}(x_r) = u_{r+1}^{(j)}(x_r), \quad r = 1, 2, \dots, N - 1,$$

this process is continued for all the steps. Consequently, in each step, we have n unknowns which can be estimated by solving the following system of n equations

$$\begin{aligned} \tilde{u}_r^{(j)}(x_r) = & f^{(j)}(x_r) + \lambda \sum_{i=1}^{r-1} \int_{x_{i-1}}^{x_i} \frac{\partial^j}{\partial x^j} k(x, t, \tilde{P}_{2n-1}^{[i]}(u; t)) \Big|_{x=x_r} dt \\ & + \lambda (\mathcal{K}_r \tilde{P}_{2n-1}^{[r]})^{(j)}(x_r), \quad j = 0, 1, \dots, n - 1, \quad r = 1, 2, \dots, N, \end{aligned} \tag{2.15}$$

where $\tilde{P}_{2n-1}^{[r]}(u; x)$, $r = 1, 2, \dots, N$, are approximations of $P_{2n-1}^{[r]}(u; x)$, $r = 1, 2, \dots, N$, and defined as

$$\begin{aligned} \tilde{P}_{2n-1}^{[r]}(u; x) = & \sum_{j=0}^{n-1} \sum_{s=0}^1 (-1)^{js} \tilde{\phi}_{n,j}^{[r]} \\ & \left((x_{r-1} + x_r)s + (-1)^s x \right) \tilde{u}^{(j)}(hs + x_{r-1}), \quad x \in \sigma_r, \quad r = 1, 2, \dots, N. \end{aligned}$$

It should be noted that, similar to (2.10) we have

$$\begin{cases} \tilde{u}^{(j)}(x_0) = u^{(j)}(x_0), \\ \tilde{u}^{(j)}(x_r) \simeq u^{(j)}(x_r), \quad r = 1, 2, \dots, N, \end{cases} \quad j = 0, 1, \dots, n - 1.$$

Finally, the N -step TTF can approximate the solution of nonlinear VIE (1.1) over the interval $[a, b]$ as

$$u(x) \simeq \mathcal{P}_{2n-1}^N(u; x) = \begin{cases} \tilde{P}_{2n-1}^{[1]}(u; x), & x \in \sigma_1, \\ \tilde{P}_{2n-1}^{[2]}(u; x), & x \in \sigma_2, \\ \vdots \\ \tilde{P}_{2n-1}^{[N]}(u; x), & x \in \sigma_N. \end{cases}$$

Note that the one-step TTF ($N = 1$) is exactly the method that was explained in the previous subsection. Also, the integrals in (2.15) can be estimated using the Gauss–Legendre quadrature formula (2.12).

3 Convergence analysis

In the current section, we study the error estimate of the TTF method in solving VIE (1.1). Let us firstly recall a discrete inequality of Gronwall type which is useful in the selected method to inspect the convergence analysis of the presented method.

Lemma 3.1 [62] *Suppose that $\{a_n\}_{n=n_0}^\infty$ and $\{b_n\}_{n=n_0}^\infty$ are two given sequences and*

$$a_n \leq c + \sum_{k=n_0}^{n-1} a_k b_k, \quad n = n_0, n_0 + 1, \dots,$$

where $n_0 \in \mathbb{N} \cup \{0\}$ and $c \geq 0$ is a real constant, also let $b_n \geq 0$ for $n = n_0, n_0 + 1, \dots$, then

$$a_n \leq c \prod_{k=n_0}^{n-1} [1 + b_k], \quad n = n_0, n_0 + 1, \dots$$

Now, consider the following linear VIE of the second kind

$$u(x) = f(x) + \lambda \int_a^x k(x, t)u(t)dt, \quad x \in I. \tag{3.1}$$

Similar to what described before for nonlinear case (1.1), to derive the TTF approximation of u , the exact solution of Eq. (3.1), we need different orders of derivatives of u at $x = a$ and $x = b$, which can be calculated as

$$u^{(j)}(x) = f^{(j)}(x) + \sum_{i=0}^{j-1} (h_{j-i-1}(x)u(x))^{(i)} + \lambda \int_a^x \frac{\partial^j k(x, t)}{\partial x^j} u(t)dt, \quad j = 0, 1, \dots, n - 1, \tag{3.2}$$

where

$$h_i(x) = \lambda \left. \frac{\partial^i k(x, t)}{\partial x^i} \right|_{t=x}.$$

Applying the general Leibniz rule, (3.2) can be reformulated as

$$u^{(j)}(x) = f^{(j)}(x) + \sum_{i=0}^{j-1} \sum_{k=0}^i H_{i,j,k}(x) u^{(k)}(x) + \lambda \int_a^x \frac{\partial^j k(x, t)}{\partial x^j} u(t) dt, \quad j = 0, 1, \dots, n - 1, \tag{3.3}$$

in which

$$H_{i,j,k}(x) = \binom{i}{k} (h_{j-i-1}(x))^{(i-k)}.$$

Equation (3.3) can be rewritten as

$$u^{(j)}(x) = f^{(j)}(x) + \sum_{k=0}^{j-1} \mathcal{H}_{i,j,k}(x) u^{(k)}(x) + \lambda \int_a^x \frac{\partial^j k(x, t)}{\partial x^j} u(t) dt, \quad j = 0, 1, \dots, n - 1, \tag{3.4}$$

where

$$\mathcal{H}_{i,j,k}(x) = \sum_{i=k}^{j-1} H_{i,j,k}(x), \quad j = 0, 1, \dots, n - 1.$$

Although $u^{(j)}(a)$, $j = 0, 1, \dots, n - 1$ can be exactly computed from (3.4), the values of $u^{(j)}(b)$, $j = 0, 1, \dots, n - 1$ are not generally available. Therefore, the TTF (2.9) is constructed by applying $\tilde{u}^{(j)}(b)$, $j = 0, 1, \dots, n - 1$, which are the obtained estimations of different orders of derivatives of u at $x = b$ by solving the following system of algebraic equations

$$\tilde{u}^{(j)}(b) = f^{(j)}(b) + \sum_{k=0}^{j-1} \mathcal{H}_{i,j,k}(b) \tilde{P}_{2n-1}^{(k)}(u; b) + \lambda \int_a^b \left. \frac{\partial^j k(x, t)}{\partial x^j} \right|_{x=b} \tilde{P}_{2n-1}(u; t) dt, \quad j = 0, 1, \dots, n - 1. \tag{3.5}$$

We can deduce from (3.4), (3.5) and interpolation conditions $\tilde{P}_{2n-1}^{(j)}(u; b) = \tilde{u}^{(j)}(b)$, $j = 0, 1, \dots, n - 1$ that

$$\begin{aligned} \left| u^{(j)}(b) - \tilde{u}^{(j)}(b) \right| &\leq \sum_{k=0}^{j-1} |\mathcal{H}_{i,j,k}(b)| \left| u^{(k)}(b) - \tilde{u}^{(k)}(b) \right| + |\lambda| \int_a^b \left| \frac{\partial^j k(x, t)}{\partial x^j} \Big|_{x=b} \right| \\ &\quad \left| u(t) - \tilde{P}_{2n-1}(u; t) \right| dt \\ &\leq \sum_{k=0}^{j-1} |\mathcal{H}_{i,j,k}(b)| \left| u^{(k)}(b) - \tilde{u}^{(k)}(b) \right| + |\lambda| \int_a^b \left| \frac{\partial^j k(x, t)}{\partial x^j} \Big|_{x=b} \right| \\ &\quad \left| u(t) - P_{2n-1}(u; t) \right| dt \\ &\quad + |\lambda| \int_a^b \left| \frac{\partial^j k(x, t)}{\partial x^j} \Big|_{x=b} \right| \left| P_{2n-1}(u; t) - \tilde{P}_{2n-1}(u; t) \right| dt, \end{aligned} \tag{3.6}$$

and subsequently, from (2.4) we have

$$\begin{aligned} \left| u^{(j)}(b) - \tilde{u}^{(j)}(b) \right| &\leq \sum_{k=0}^{j-1} |\mathcal{H}_{i,j,k}(b)| \left| u^{(k)}(b) - \tilde{u}^{(k)}(b) \right| + |\lambda| (b - a) \\ &\quad M_n \left(E_n + \left\| P_{2n-1}(u; t) - \tilde{P}_{2n-1}(u; t) \right\|_\infty \right), \end{aligned}$$

where

$$M_n = \max_{0 \leq j \leq n-1} \left\{ \max_{a \leq x, t \leq b} \left\{ \left| \frac{\partial^j k(x, t)}{\partial x^j} \right| \right\} \right\}, \quad E_n = \frac{(b - a)^{2n} \|u^{(2n)}(t)\|_\infty}{2^{2n} (2n)!}. \tag{3.7}$$

Thus, Lemma 3.1 implies

$$\begin{aligned} \left| u^{(j)}(b) - \tilde{u}^{(j)}(b) \right| &\leq |\lambda| (b - a) M_n \left(E_n + \left\| P_{2n-1}(u; t) - \tilde{P}_{2n-1}(u; t) \right\|_\infty \right) \\ &\quad \prod_{k=0}^{j-1} [1 + |\mathcal{H}_{i,j,k}(b)|]. \end{aligned} \tag{3.8}$$

This result allows us to derive an upper bound to the error of the proposed method for solving linear VIE (3.1).

Theorem 3.1 *Let $P_{2n-1}(u; x)$ obtained by (2.5) be the polynomial interpolant of the exact solution of linear VIE (3.1), and let polynomial $\tilde{P}_{2n-1}(u; x)$ defined in (2.9) be its approximation. Also suppose that the following condition holds:*

$$m_n \Delta_n M_n (b - a) < 1,$$

where

$$m_n = \max_{a \leq x \leq b} \left\{ \sum_{j=0}^{n-1} \tilde{\phi}_{n,j}(x) \right\}, \quad \Delta_n = \max_{0 \leq j \leq n-1} \left\{ \prod_{k=0}^{j-1} [1 + |\mathcal{H}_{i,j,k}(b)|] \right\},$$

and M_n is as introduced before. Then, we have

$$\left\| P_{2n-1}(u; x) - \tilde{P}_{2n-1}(u; x) \right\|_{\infty} \leq \frac{m_n \Delta_n M_n E_n |\lambda|(b-a)}{1 - m_n \Delta_n M_n |\lambda|(b-a)}, \tag{3.9}$$

where E_n is as introduced in (3.7).

Proof Since $u^{(j)}(a)$, $j = 0, 1, \dots, n - 1$, can be calculated exactly by substituting $x = a$ into (3.4), from (2.5), (2.9) and (3.8) we get

$$\begin{aligned} \left| P_{2n-1}(u; x) - \tilde{P}_{2n-1}(u; x) \right| &\leq \sum_{j=0}^{n-1} \tilde{\phi}_{n,j}(a+b-x) \left| u^{(j)}(b) - \tilde{u}^{(j)}(b) \right| \\ &\leq |\lambda|(b-a) M_n \left(E_n + \left\| P_{2n-1}(u; t) - \tilde{P}_{2n-1}(u; t) \right\|_{\infty} \right) \\ &\quad \prod_{k=0}^{j-1} [1 + |\mathcal{H}_{i,j,k}(b)|] \sum_{j=0}^{n-1} \tilde{\phi}_{n,j}(a+b-x). \end{aligned} \tag{3.10}$$

The symmetry of $\tilde{\phi}_{n,j}(x)$ and $\tilde{\phi}_{n,j}(a+b-x)$ with respect to $x = \frac{a+b}{2}$ implies that

$$m_n = \max_{a \leq x \leq b} \left\{ \sum_{j=0}^{n-1} \tilde{\phi}_{n,j}(a+b-x) \right\}.$$

Subsequently, it can be deduced from (3.10) that

$$\begin{aligned} \left| P_{2n-1}(u; x) - \tilde{P}_{2n-1}(u; x) \right| &\leq m_n \Delta_n M_n |\lambda|(b-a) \\ &\quad \left(E_n + \left\| P_{2n-1}(u; t) - \tilde{P}_{2n-1}(u; t) \right\|_{\infty} \right), \end{aligned}$$

and therefore

$$(1 - m_n \Delta_n M_n |\lambda|(b-a)) \left\| P_{2n-1}(u; x) - \tilde{P}_{2n-1}(u; x) \right\|_{\infty} \leq m_n \Delta_n M_n E_n |\lambda|(b-a),$$

which implies (3.9). □

Theorem 3.2 Suppose that $u \in C^{2n}[a, b]$ is the exact solution of linear VIE (3.1). Then, under the assumptions of Theorem 3.1, we have

$$\|u(x) - \tilde{P}_{2n-1}(u; x)\|_{\infty} \leq \frac{E_n}{1 - m_n \Delta_n M_n |\lambda|(b-a)},$$

where E_n has been introduced in (3.7).

Proof Corollary 2.1 and Theorem 3.1 imply that

$$\begin{aligned} \|u(x) - \tilde{P}_{2n-1}(u; x)\|_\infty &\leq \|u(x) - P_{2n-1}(u; x)\|_\infty \\ &\quad + \|P_{2n-1}(u; x) - \tilde{P}_{2n-1}(u; x)\|_\infty \\ &\leq \frac{\|u^{(2n)}(x)\|_\infty}{(2n)!} \|(x-a)^n(b-x)^n\|_\infty \\ &\quad + \frac{m_n \Delta_n M_n E_n |\lambda|(b-a)}{1 - m_n \Delta_n M_n |\lambda|(b-a)} \\ &\leq E_n \left(1 + \frac{m_n \Delta_n M_n |\lambda|(b-a)}{1 - m_n \Delta_n M_n |\lambda|(b-a)} \right) \\ &= \frac{E_n}{1 - m_n \Delta_n M_n |\lambda|(b-a)}. \end{aligned}$$

□

Remark 3.1 It should be emphasized that the error analysis for nonlinear VIE (1.1) is considerably more complicated than linear case (3.1). Nevertheless, in what follows, we try to present the outline of the error estimate of the proposed method for nonlinear VIEs. For $j = 0, 1, \dots, n - 1$, we can get from (2.6) and (2.8) that

$$\begin{aligned} \left| u^{(j)}(b) - \tilde{u}^{(j)}(b) \right| &= |\lambda| \left| (\mathcal{K}u)^{(j)}(b) - (\mathcal{K}\tilde{P}_{2n-1})^{(j)}(b) \right| \\ &\leq |\lambda| \sum_{i=0}^{j-1} \left| (\mathcal{D}_{j-i-1}u)^{(i)}(b) - (\mathcal{D}_{j-i-1}\tilde{P}_{2n-1})^{(i)}(b) \right| \quad (3.11) \\ &\quad + |\lambda| \int_a^b \left| \frac{\partial^j}{\partial x^j} \left(k(x, t, u(t)) - k(x, t, \tilde{P}_{2n-1}(u; t)) \right) \right|_{x=b} dt. \end{aligned}$$

Note that $(\mathcal{D}_j u)(x)$ is generally a function of x and $u(x)$. Therefore, the i -th derivative of this function with respect to x , which we denote by $(\mathcal{D}_j u)^{(i)}(x)$, includes x , $u(x)$, and the first i derivatives of u . Now, suppose that $(\mathcal{D}_j u)(x)$, $j = 0, 1, \dots, n - 1$ and their derivatives satisfy the following Lipschitz conditions

$$\left| (\mathcal{D}_j u_1)^{(i)}(x) - (\mathcal{D}_j u_2)^{(i)}(x) \right| \leq \sum_{k=0}^i L_k \left| u_1^{(k)} - u_2^{(k)} \right|, \quad i = 0, 1, \dots, n - 2. \quad (3.12)$$

Also suppose that the kernel $k : [a, b]^2 \times \mathbb{R} \rightarrow \mathbb{R}$ and its derivatives are Lipschitz continuous in u , that is

$$\left| \frac{\partial^j}{\partial x^j} k(x, t, u_1) - \frac{\partial^j}{\partial x^j} k(x, t, u_2) \right| \leq \bar{L}_j |u_1 - u_2|, \quad j = 0, 1, \dots, n - 1. \tag{3.13}$$

Then inequality (3.11) implies that

$$\begin{aligned} \left| u^{(j)}(b) - \tilde{u}^{(j)}(b) \right| &\leq L |\lambda| \sum_{i=0}^{j-1} \sum_{k=0}^i \left| u^{(k)}(b) - \tilde{u}^{(k)}(b) \right| \\ &\quad + \bar{L} |\lambda| \int_a^b \left| u(t) - \tilde{P}_{2n-1}(u; t) \right| dt \\ &\leq L |\lambda| \sum_{k=0}^{j-1} (j - k) \left| u^{(k)}(b) - \tilde{u}^{(k)}(b) \right| \\ &\quad + \bar{L} |\lambda| (b - a) \left\| u(t) - \tilde{P}_{2n-1}(u; t) \right\|_{\infty}, \end{aligned}$$

where $L = \max_{0 \leq k \leq n-2} \{L_k\}$ and $\bar{L} = \max_{0 \leq j \leq n-1} \{\bar{L}_j\}$. The continuation of the procedure is similar to the linear case.

We present now the matrix form of system (2.11) in order to investigate the existence and uniqueness of its solution. Let us define

$$\begin{aligned} U &= \left[\tilde{u}(b), \tilde{u}'(b), \dots, \tilde{u}^{(n-1)}(b) \right]^T, \\ F_0 &= \left[f(b), f'(b), \dots, f^{(n-1)}(b) \right]^T, \\ D(U) &= [d_0, d_1, \dots, d_{n-1}]^T, \\ G(U) &= [g_0, g_1, \dots, g_{n-1}]^T, \end{aligned}$$

where

$$\begin{cases} d_0 = 0, \\ d_j = \sum_{i=0}^{j-1} \left(\mathcal{D}_{j-i-1} \tilde{P}_{2n-1} \right)^{(i)}(b), \quad j = 1, 2, \dots, n - 1, \end{cases}$$

and

$$g_j = \int_a^b \frac{\partial^j}{\partial x^j} k(x, t, \tilde{P}_{2n-1}(u; t)) \Big|_{x=b} dt, \quad j = 0, 1, \dots, n - 1.$$

Then, system (2.11) can be formulated in the matrix form

$$U = F_0 + \lambda (D(U) + G(U)),$$

which can be solved by using the following iterative method

$$U^{[v+1]} = F_0 + \lambda (D(U^{[v]}) + G(U^{[v]})), \quad v = 0, 1, \dots, \tag{3.14}$$

with the starting value $U^{[0]}$.

Remark 3.2 Suppose that the Lipschitz conditions (3.12) and (3.13) are satisfied and the iterative procedure (3.14) is utilized to solve system (2.11). We can get from (3.14) that

$$\begin{aligned} \|U^{[v+1]} - U^{[v]}\|_\infty &= |\lambda| \| (D(U^{[v]}) - D(U^{[v-1]})) + (G(U^{[v]}) - G(U^{[v-1]})) \|_\infty \\ &\leq |\lambda| (\|D(U^{[v]}) - D(U^{[v-1]})\|_\infty + \|G(U^{[v]}) - G(U^{[v-1]})\|_\infty). \end{aligned} \tag{3.15}$$

The Lipschitz condition (3.12) and the interpolation conditions $\tilde{P}_{2n-1}^{(j)}(u; t) = \tilde{u}^{(j)}(b), j = 0, 1, \dots, n - 1$ imply that each element of the vector $D(U^{[v]}) - D(U^{[v-1]})$ satisfies the following inequality:

$$\begin{aligned} |d_j^{[v]} - d_j^{[v-1]}| &\leq \sum_{i=0}^{j-1} \left| \left(\mathcal{D}_{j-i-1} \tilde{P}_{2n-1}^{[v]} \right)^{(i)}(b) - \left(\mathcal{D}_{j-i-1} \tilde{P}_{2n-1}^{[v-1]} \right)^{(i)}(b) \right| \\ &\leq \sum_{i=0}^{j-1} \sum_{k=0}^i L_k \left| \left(\tilde{P}_{2n-1}^{[v]} \right)^{(k)}(b) - \left(\tilde{P}_{2n-1}^{[v-1]} \right)^{(k)}(b) \right| \\ &= \sum_{i=0}^{j-1} \sum_{k=0}^i L_k \left| \left(\tilde{u}^{(k)} \right)^{[v]}(b) - \left(\tilde{u}^{(k)} \right)^{[v-1]}(b) \right| \\ &= \sum_{k=0}^{j-1} (j-k) L_k \left| \left(\tilde{u}^{(k)} \right)^{[v]}(b) - \left(\tilde{u}^{(k)} \right)^{[v-1]}(b) \right|, \\ &j = 0, 1, \dots, n - 1. \end{aligned} \tag{3.16}$$

Since generally we have

$$\left| \left(\tilde{u}^{(k)} \right)^{[v]}(b) - \left(\tilde{u}^{(k)} \right)^{[v-1]}(b) \right| \leq \|U^{[v]} - U^{[v-1]}\|_\infty, \quad k = 0, 1, \dots, j - 1,$$

inequality (3.16) signifies that

$$\begin{aligned} |d_j^{[v]} - d_j^{[v-1]}| &\leq \sum_{k=0}^{j-1} (j-k) L_k \|U^{[v]} - U^{[v-1]}\|_\infty \\ &\leq \frac{j(j+1)}{2} L \|U^{[v]} - U^{[v-1]}\|_\infty, \quad j = 0, 1, \dots, n - 1, \end{aligned}$$

where L is as introduced before. Therefore, we can write

$$\|D(U^{[v]}) - D(U^{[v-1]})\|_\infty \leq \frac{n(n-1)}{2}L \|U^{[v]} - U^{[v-1]}\|_\infty. \tag{3.17}$$

Furthermore, from the Lipschitz condition (3.13), it can be concluded that

$$\|G(U^{[v]}) - G(U^{[v-1]})\| \leq \bar{L}(b-a) \|\tilde{P}_{2n-1}^{[v]}(u; t) - \tilde{P}_{2n-1}^{[v-1]}(u; t)\|_\infty. \tag{3.18}$$

As mentioned before, the polynomial $\tilde{P}_{2n-1}(u; t)$ is constructed by using the exact values $u^{(j)}(a)$ and the approximate values $\tilde{u}^{(j)}(b)$ for $j = 0, 1, \dots, n - 1$. Thus, by introducing

$$\mathcal{E}(t) = \left[\tilde{\phi}_{n,0}(a + b - t), -\tilde{\phi}_{n,1}(a + b - t), \dots, (-1)^{n-1} \tilde{\phi}_{n,n-1}(a + b - t) \right]^T,$$

we can deduce

$$\begin{aligned} \left| \tilde{P}_{2n-1}^{[v]}(u; t) - \tilde{P}_{2n-1}^{[v-1]}(u; t) \right| &= \left| \mathcal{E}^T(t) (U^{[v]} - U^{[v-1]}) \right| \\ &\leq n \|\mathcal{E}^T(t)\|_\infty \|U^{[v]} - U^{[v-1]}\|_\infty, \end{aligned}$$

which together with (3.15), (3.17) and (3.18) yields

$$\begin{aligned} \|U^{[v+1]} - U^{[v]}\|_\infty &\leq |\lambda| \left(\frac{n(n-1)}{2}L \|U^{[v]} - U^{[v-1]}\|_\infty \right. \\ &\quad \left. + n\bar{L}(b-a) \|\mathcal{E}^T(t)\|_\infty \|U^{[v+1]} - U^{[v]}\|_\infty \right). \end{aligned}$$

Therefore, if $|\lambda| \left(\frac{n(n-1)}{2}L + n\bar{L}(b-a) \|\mathcal{E}^T(t)\|_\infty \right) < 1$, then it follows from the contraction mapping theorem [63] that the iterative method (3.14) has a unique solution.

The convergence of the proposed method for solving both the linear and nonlinear VIEs is analysed numerically in the next section.

4 Illustrative examples

In this section, the precision of the presented TTF and its multistep form for solving integral equation (1.1) is investigated during the numerical experiments. In order to ensure the competence of the method, we applied it to solve several VIEs. The obtained results for nine examples are reported in this section. To confirm the accuracy of the method, we compute and report the absolute errors of the approximations. Also, the obtained results are compared with the results of the other existing methods in the literature. We employ the TTF method for Example 4.1, both the TTF and multistep

TTF methods for Examples 4.2–4.6, and only the multistep TTF method for Examples 4.7–4.9. The accuracy of the multistep TTF for the examples is investigated by computing the L_∞ error, which is defined as

$$L_\infty = \left\| u_{ex}(x) - \mathcal{P}_{2n-1}^N(u; x) \right\|_\infty,$$

where $u_{ex}(x)$ is the exact solution of each problem. In order to increase the speed of the algorithm, the Gauss–Legendre quadrature rule is utilized for estimating the integrals which appeared in the method. In the reported results, the quadrature formula is constructed by taking $m = 15$ for Examples 4.2, 4.3, 4.4, and $m = 10$ for the other examples. The used computational time for each problem is also reported. In order to solve systems (2.11) and (2.15) for the examples, we utilize the `fsolve` command in Maple software. All the computations are written in Maple 2016 software and run on a Core (TM) i7 PC with 2.70 GHz of CPU and 16 GB of RAM.

Example 4.1 As the first example, consider the nonlinear VIE [64]

$$u(x) = 2 - \cos(x) - \int_0^x \sin(xu(t) - t) dt, \quad x \in [0, 5].$$

with the exact solution $u_{ex}(x) = 1$,

Taking $n = 1$, the TTF approximates the exact solution of this problem as

$$\tilde{P}_1(u; x) = 1 - 6.0550704156 \times 10^{-20}x,$$

which implies that the method can calculate the exact solution of VIEs with polynomial solutions. The result has been obtained in only 0.359 s of computational time.

Example 4.2 Now, consider the nonlinear problem [17,18]

$$u(x) = \sin(x) - e^{\sin(x)} + 1 + \int_0^x \cos(t)e^{u(t)} dt, \quad x \in [0, 1],$$

with the exact solution $u_{ex}(x) = \sin(x)$.

We have utilized the TTF with $n = 10$ to solve this problem. The exact and approximate solutions and the absolute error of the TTF for solving this example are plotted in Fig. 1. As can be seen, the TTF gives accurate approximation to the solution of this problem over interval $[0, 1]$. Table 1 is provided to compare the TTF results with the reported results of LCQM [17] and NPSQM [18]. It indicates the accuracy of the proposed method in comparison to the other two methods. The used CPU time in this case is 1.248 s. We have also utilized the multistep TTF to obtain an appropriate estimation of the solution over large intervals. The obtained L_∞ errors and the used CPU times are reported in Table 2. The table shows the acceptable speed and accuracy of the method over large intervals and the influence of increasing n on the precision of the approximation.

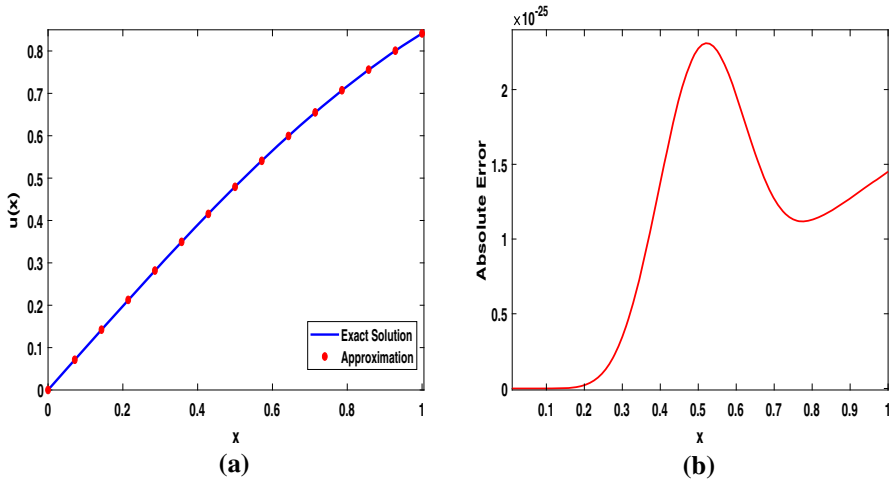


Fig. 1 a Comparison between the exact and approximate solutions for Example 4.2, b the absolute error of the TTF for Example 4.2

Table 1 Absolute errors for Example 4.2

x	LCQM [17]	NPSQM [18]	TTF
0.1	6.23e-06	7.6e-12	6.88e-30
0.2	4.63e-06	2.9e-11	2.22e-27
0.3	6.21e-06	7.1e-11	3.46e-26
0.4	1.43e-06	1.4e-10	1.38e-25
0.5	7.01e-06	2.4e-10	2.27e-25
0.6	3.68e-06	3.8e-10	1.96e-25
0.7	4.22e-06	5.7e-10	1.27e-25
0.8	4.77e-06	7.9e-10	1.13e-25
0.9	5.64e-07	1.0e-09	1.27e-25

Table 2 Results of the multistep TTF with $N = 10$ for Example 4.2

n	$[a, b] = [0, 10]$		$[a, b] = [0, 20]$	
	L_∞	CPU time (s)	L_∞	CPU time (s)
4	1.60e-07	1.953	4.00e-05	1.891
6	8.00e-13	2.782	2.50e-09	2.766
8	1.00e-18	4.500	2.00e-13	4.453

Example 4.3 In this example, we consider the following nonlinear VIE in the complex plane [65]

$$u(x) = f_1(x) + if_2(x) + \frac{1}{50} \int_0^x x^2 t^2 u^2(t) dt, \quad x \in [0, 1],$$

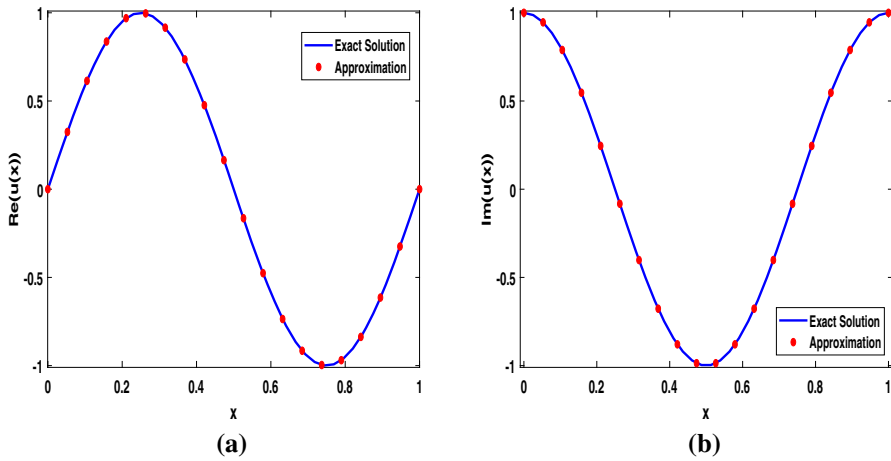


Fig. 2 Comparison between **a** the real, **b** the imaginary parts of the exact and approximate solutions for Example 4.3

where

$$f_1(x) = \sin(2\pi x) + \frac{x^2 (8\pi^2 x^2 \cos(2\pi x) \sin(2\pi x) + 4\pi x \cos^2(2\pi x) - 2\pi x - \cos(2\pi x) \sin(2\pi x))}{800\pi^3},$$

$$f_2(x) = \cos(2\pi x) + \frac{x^2 (8\pi^2 x^2 \cos^2(2\pi x) - 4\pi^2 x^2 - 4\pi x \cos(2\pi x) \sin(2\pi x) + \sin^2(2\pi x))}{800\pi^3}.$$

The exact solution of this problem is

$$u_{ex}(x) = \sin(2\pi x) + i \cos(2\pi x).$$

The TTF by using $n = 15$ is determined and its real and imaginary parts are compared with the exact solution in Fig. 2. In order to separately investigate the accuracy of the real and imaginary parts of the approximation, we define the following error functions

$$E_1(x) = \left| \text{Re} \left(\tilde{P}_{2n-1}(u; x) \right) - \sin(2\pi x) \right|, \quad E_2(x) = \left| \text{Im} \left(\tilde{P}_{2n-1}(u; x) \right) - \cos(2\pi x) \right|.$$

These functions are displayed in Fig. 3. Also the absolute error which is defined as

$$\text{Absolute Error} = \sqrt{(E_1(x))^2 + (E_2(x))^2},$$

is calculated in some selected points and compared with the results of the RH wavelet method (RHW) [65] in Table 3. The used CPU times of the proposed method and the RHW are also compared in this table. In order to investigate the accuracy of the multistep TTF for solving this problem, we have employed it by taking $N = 30$ over intervals $[0, 1]$ and $[0, 5]$. The obtained L_∞ errors and the used CPU times are

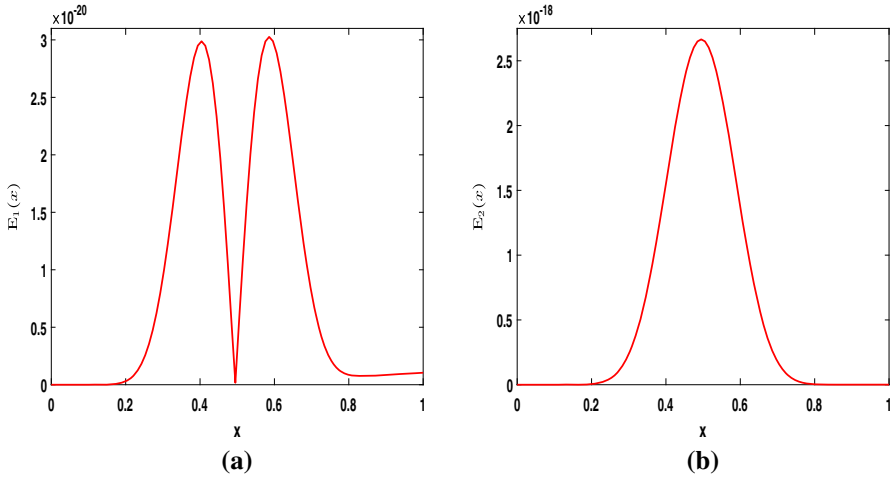


Fig. 3 The error of approximating **a** the real, **b** the imaginary part of the solution for Example 4.3

Table 3 Absolute errors for Example 4.3

x	RHWM [65]	TTF
0.1	2.91e-15	5.87e-25
0.2	4.49e-12	3.29e-21
0.3	5.05e-12	1.95e-19
0.4	3.29e-09	1.44e-18
0.5	1.30e-08	2.67e-18
0.6	2.57e-07	1.44e-18
0.7	6.86e-07	1.95e-19
0.8	5.74e-07	3.88e-21
0.9	6.91e-06	1.05e-21
CPU time (s)	3.172	1.841

Table 4 Results of the multistep TTF with $N = 30$ for Example 4.3

n	$[a, b] = [0, 1]$		$[a, b] = [0, 5]$	
	L_∞	CPU time (s)	L_∞	CPU time (s)
3	1.80e-09	5.094	3.00e-04	4.672
5	4.00e-17	15.484	6.00e-09	11.375
7	2.00e-25	29.797	1.80e-14	27.031

reported in Table 4. It can be concluded from the results that the TTF and its multistep form are dependable methods for solving VIEs in the complex plane.

Example 4.4 Consider the following VIE of the first kind [66]

$$\int_0^x \frac{e^{x-t}}{1+x^2} u(t) dt = -\frac{4\pi \cos(4\pi x) + \sin(4\pi x) - 4\pi e^x}{(1+x^2)(1+16\pi^2)}, \quad (4.1)$$

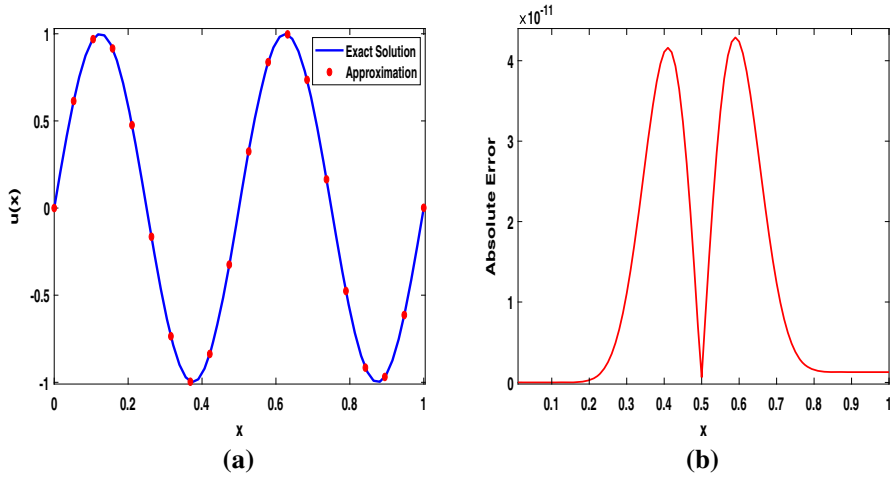


Fig. 4 **a** Comparison between the exact and approximate solutions for Example 4.4, **b** the absolute error of the TTF for Example 4.4

where the exact solution is

$$u_{ex}(x) = \sin(4\pi x).$$

Applying differentiation with respect to x on both sides of (4.1), the following second kind VIE is obtained:

$$u(x) = \frac{-1}{1 + 16\pi^2} \left(-16\pi^2 \sin(4\pi x) + 4\pi \cos(4\pi x) - 4\pi e^x \right) - \int_0^x e^{x-t} u(t) dt, \quad x \in [0, 1].$$

Now, the TTF (2.9) can be used to approximate the solution of this problem. The exact and approximate solutions and the absolute error of the TTF by using $n = 15$ are displayed in Fig. 4. Also, we compare the TTF with the Haar wavelets method (HWM) [66], Daubechies wavelets method (DWM) [66], and Coifman wavelets method (CWM) [66] in Table 5. The listed absolute errors and computational times indicate that the proposed TTF is fast and can achieve more accurate results in comparison to the other mentioned methods. Also, the multistep TTF with $N = 10$ and different choices of n has been utilized to estimate the solution of this problem over intervals $[0, 1]$ and $[0, 5]$. The obtained results are reported in Table 6. This table shows that by choosing suitable values of n and N , the method can provide acceptable results in short computational times.

Table 5 Absolute errors for Example 4.4

x	HWM [66]	DWM [66]	CWM [66]	TTF
0.0	9.8e-02	7.3e-04	4.1e-04	0.00
0.2	4.7e-02	4.9e-04	3.0e-04	2.79e-13
0.4	6.1e-03	3.4e-05	1.5e-05	4.11e-11
0.6	5.7e-03	3.1e-04	1.7e-04	4.24e-11
0.8	4.7e-02	6.3e-04	3.5e-04	1.56e-12
1.0	0.00	7.5e-04	4.0e-04	1.28e-12
CPU time (s)	57	227	25	2.294

Table 6 Results of the multistep TTF with $N = 10$ for Example 4.4

n	$[a, b] = [0, 1]$		$[a, b] = [0, 5]$	
	L_∞	CPU time (s)	L_∞	CPU time (s)
6	7.00e-12	7.266	6.00e-05	7.078
8	2.50e-17	12.828	1.00e-07	12.657
10	3.00e-23	21.797	6.00e-11	20.688

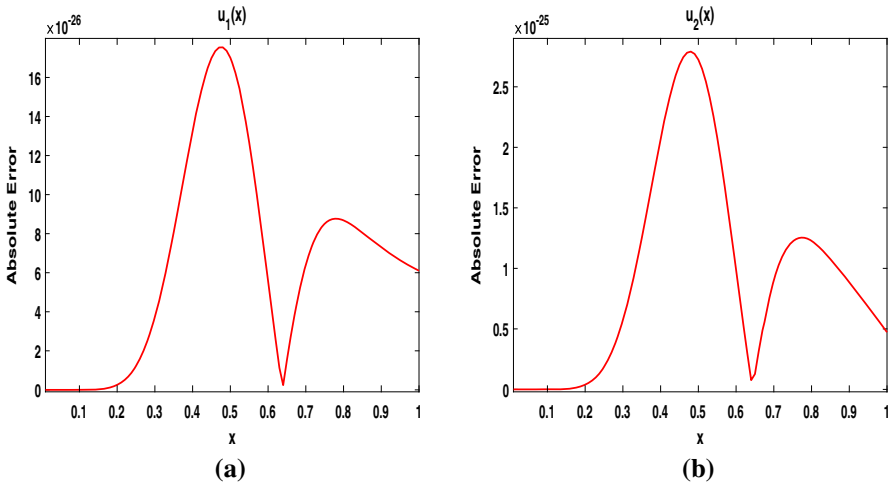


Fig. 5 The absolute error of **a** $u_1(x)$, **b** $u_2(x)$ for Example 4.5

Example 4.5 Now, consider the following system of linear VIEs of the second kind [67,68]

$$\begin{cases} u_1(x) = \cosh(x) + x \sin(x) - \int_0^x (e^{x-t}u_1(t) + \cos(x-t)u_2(t)) dt, \\ u_2(x) = 2 \sin(x) + x (\sin^2(x) + e^x) - \int_0^x (e^{x+t}u_1(t) + x \cos(t)u_2(t)) dt, \end{cases} \quad x \in [0, 1],$$

Table 7 Absolute errors for Example 4.5

x	BSM [67]		BBM [68]		TTF	
	$u_1(x)$	$u_2(x)$	$u_1(x)$	$u_2(x)$	$u_1(x)$	$u_2(x)$
0.1	3.73e-05	6.52e-06	1.20e-09	3.80e-09	8.30e-30	1.23e-29
0.2	7.82e-05	1.77e-05	2.70e-09	8.40e-09	2.55e-27	3.85e-27
0.3	1.12e-04	1.74e-05	2.50e-09	1.40e-08	3.73e-26	5.72e-26
0.4	1.36e-04	1.37e-05	3.20e-09	2.17e-08	1.33e-25	2.07e-25
0.5	1.62e-04	8.60e-05	3.30e-09	2.43e-08	1.70e-25	2.72e-25
0.6	3.17e-04	8.54e-05	3.00e-09	3.40e-08	5.56e-26	9.85e-26
0.7	8.21e-04	8.84e-04	2.70e-09	4.40e-08	6.40e-26	8.98e-26
0.8	2.63e-03	4.40e-03	5.00e-10	5.30e-08	8.68e-26	1.23e-25
0.9	8.80e-03	1.82e-02	4.80e-09	6.50e-08	7.30e-26	8.85e-26
1	2.81e-02	6.71e-02	4.50e-09	7.90e-08	6.10e-26	4.72e-26

Table 8 L_∞ errors of the multistep TTF with $N = 4$ for Example 4.5

n	$[a, b] = [0, 5]$			$[a, b] = [0, 10]$		
	$u_1(x)$	$u_2(x)$	CPU time (s)	$u_1(x)$	$u_2(x)$	CPU time (s)
8	5.00e-15	2.50e-13	7.063	1.60e-13	1.00e-09	6.828
10	2.50e-19	1.00e-17	10.531	1.50e-17	1.20e-14	10.859
12	4.00e-17	1.80e-15	16.140	1.60e-19	2.00e-15	16.203

with the exact solutions $u_1(x) = e^{-x}$ and $u_2(x) = 2 \sin(x)$. The absolute errors of the TTF by taking $n = 10$ for approximating $u_1(x)$ and $u_2(x)$ are plotted in Fig. 5. The TTF results are compared with the results of the biorthogonal systems method (BSM) [67] and the block by block method (BBM) [68] in Table 7. The obtained results reveal the acceptable agreement between the TTF approximations and the exact solutions of the problem over interval $[0, 1]$. The used CPU time to earn these results is only 1.560 s. Taking $N = 4$, the multistep TTF has been utilized to approximate the solution of the problem over intervals $[0, 5]$ and $[0, 10]$. The L_∞ errors of approximating $u_1(x)$ and $u_2(x)$ and also the used computational times are presented in Table 8. As expected, the accuracy of the results increases by increasing n .

Example 4.6 Consider the following system of nonlinear VIEs of the second kind [68,69]

$$\begin{cases} u_1(x) = \cos(x) - \frac{1}{2} \sin^2(x) + \int_0^x u_1(t)u_2(t)dt, \\ u_2(x) = \sin(x) - x + \int_0^x (u_1^2(t) + u_2^2(t)) dt, \end{cases} \quad x \in [0, 1],$$

with the exact solutions $u_1(x) = \cos(x)$ and $u_2(x) = \sin(x)$.

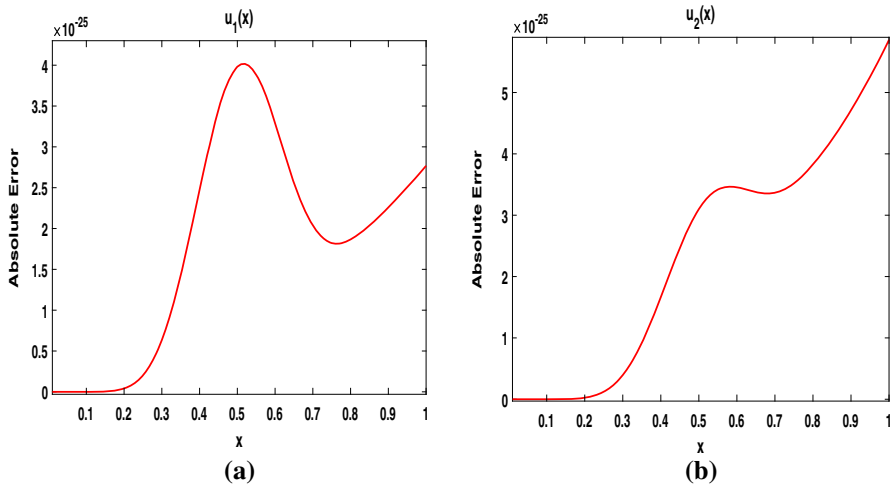


Fig. 6 The absolute error of **a** $u_1(x)$, **b** $u_2(x)$ for Example 4.6

Table 9 Absolute errors for Example 4.6

x	STWSM [69]		BBM [68]		TTF	
	$u_1(x)$	$u_2(x)$	$u_1(x)$	$u_2(x)$	$u_1(x)$	$u_2(x)$
0.1	8.31e-08	1.67e-06	9.00e-10	8.70e-09	1.30e-29	7.45e-30
0.2	3.30e-07	3.32e-06	3.30e-09	1.77e-08	4.13e-27	2.45e-27
0.3	7.33e-07	4.97e-06	7.30e-09	2.71e-08	6.34e-26	3.94e-26
0.4	1.28e-06	6.59e-06	1.29e-08	3.74e-08	2.47e-25	1.66e-25
0.5	1.95e-06	8.20e-06	1.99e-08	4.90e-08	3.98e-25	3.10e-25
0.6	2.73e-06	9.79e-06	2.84e-08	6.22e-08	3.29e-25	3.46e-25
0.7	3.58e-06	1.14e-05	3.85e-08	7.81e-08	2.03e-25	3.37e-25
0.8	4.47e-06	1.30e-05	4.96e-08	9.66e-08	1.86e-25	3.82e-25
0.9	5.36e-06	1.47e-05	6.21e-08	1.20e-07	2.26e-25	4.71e-25
1	6.22e-06	1.65e-05	7.60e-08	1.47e-07	2.77e-25	5.86e-25

We have employed the TTF with $n = 10$ to approximate the solutions of this problem. The obtained absolute errors of the method for estimating $u_1(x)$ and $u_2(x)$ are plotted in Fig. 6. The TTF results are compared with the results of the single-term Walsh series method (STWSM) [69] and the BBM [68] in Table 9. The used computational time in this case is 5.086 s. We have also implemented the multistep TTF with $N = 15$ to obtain accurate approximations to the solutions of this problem over intervals $[0, 5]$ and $[0, 10]$. The L_∞ errors and the used computational times for three choices of n are presented in Table 10. The reported results confirm the acceptable accuracy and speed of the algorithm over large intervals.

Table 10 L_∞ errors of the multistep TTF with $N = 15$ for Example 4.6

n	$[a, b] = [0, 5]$			$[a, b] = [0, 10]$		
	$u_1(x)$	$u_2(x)$	CPU time (s)	$u_1(x)$	$u_2(x)$	CPU time (s)
4	1.60e-09	2.00e-09	4.078	5.00e-05	7.00e-05	4.032
6	9.00e-17	1.00e-16	7.047	4.00e-11	6.00e-11	7.157
8	1.40e-24	1.80e-24	13.047	1.00e-17	1.60e-17	13.265

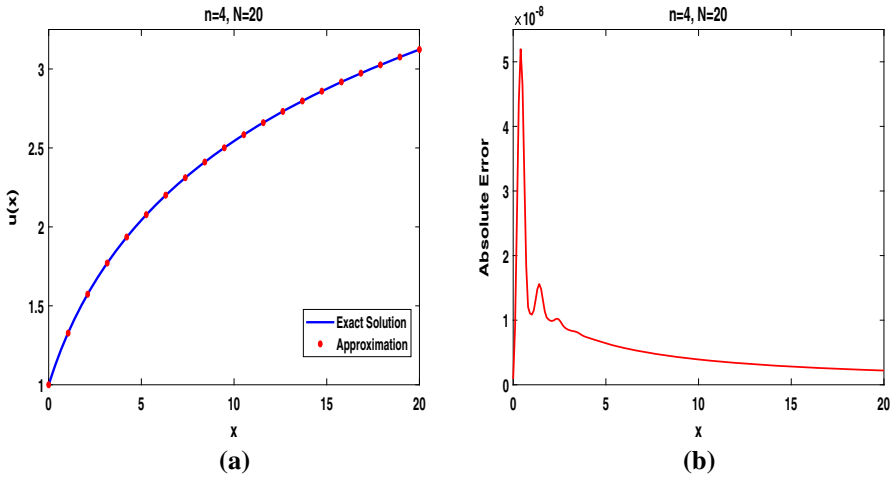


Fig. 7 **a** Comparison between the exact and approximate solutions for Example 4.7 and **b** the absolute error of the multistep TTF for Example 4.7

Example 4.7 Now, consider the nonlinear VIE [70]

$$u(x) = e^{-x} + \int_0^x e^{t-x} (u(t) + e^{-u(t)}) dt, \quad x \in [0, 20],$$

with the exact solution $u_{ex}(x) = \ln(x + e)$.

Since the problem is defined over a large interval, we employ the multistep TTF with $n = 4$ and $N = 20$ steps to approximate its solution. The exact and approximate solutions and the absolute error of the presented method are plotted in Fig. 7. This figure shows that by applying the multistep TTF, we can accurately approximate the solution of this problem over $[0, 20]$. The used CPU time in this case is 8.705 s. Also, the influence of increasing n and N on the precision of the approximation has been investigated by computing the L_∞ error of the method. The results are reported in Table 11. This table shows clearly that how we can get better results by increasing n or N .

Table 11 L_∞ errors of the multistep TTF with different values of n and N for Example 4.7

n	N			
	4	6	8	10
10	5.0e-06	2.0e-08	9.0e-11	4.0e-13
20	5.0e-08	2.0e-11	9.0e-15	4.1e-18
30	2.5e-09	2.5e-13	2.5e-17	3.0e-21
40	3.0e-10	1.0e-14	4.0e-19	1.6e-23

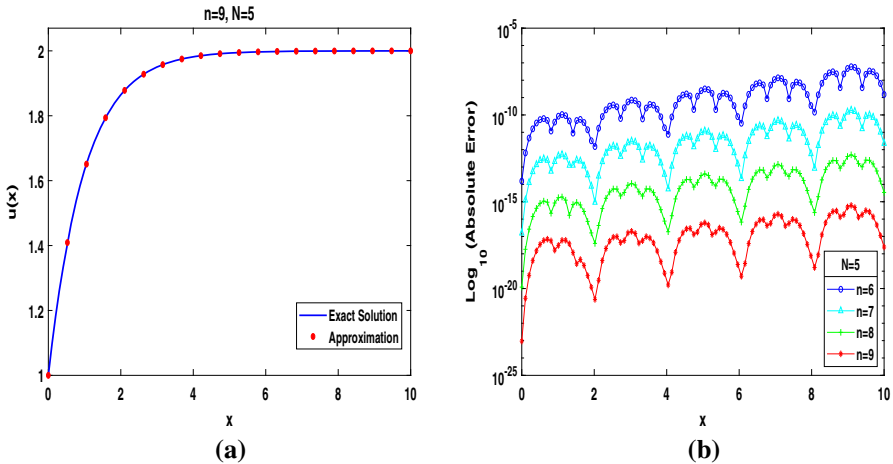


Fig. 8 **a** Comparison between the exact and approximate solutions for Example 4.8 and **b** log-plots of absolute errors of the multistep TTF for Example 4.8

Example 4.8 In this example, we consider the stiff VIE [71]

$$u(x) = f(x) - \int_0^x (100 + 1500(x - t)) u^3(t) dt, \quad x \in [0, 10],$$

where f has been chosen so that the exact solution of the problem is

$$u_{ex}(x) = 2 - e^{-x}.$$

The multistep TTF with $n = 9$ and $N = 5$ has been utilized to approximate the solution of this problem. The used computational time in this case is 20.857 s. The agreement between the obtained approximation and the theoretical solution is displayed in Fig. 8a. The Log-plots of the absolute errors of the presented method with $N = 5$ and different choices of n over $[0, 10]$ are plotted in Fig. 8b. This figure indicates that we can reduce the error of the approximation by increasing n . In Table 12, we present the absolute values of the error of the multistep TTF with a fixed n and three choices of N to confirm the influence of increasing N on the precision of the approximation.

Table 12 Absolute errors of the multistep TTF with $n = 5$ for Example 4.8

x	$N = 5$	$N = 10$	$N = 15$
0	0.00	0.00	0.00
1	5.19e-09	5.05e-12	1.05e-12
2	1.13e-09	1.71e-11	6.98e-13
3	8.82e-08	4.79e-11	7.92e-12
4	6.42e-09	1.36e-10	6.36e-12
5	4.17e-07	3.86e-10	7.20e-11
6	3.06e-08	1.09e-09	5.76e-11
7	1.99e-06	3.09e-09	6.53e-10
8	1.46e-07	8.75e-09	5.22e-10
9	9.49e-06	2.48e-08	5.92e-09
10	6.96e-07	7.01e-08	4.74e-09

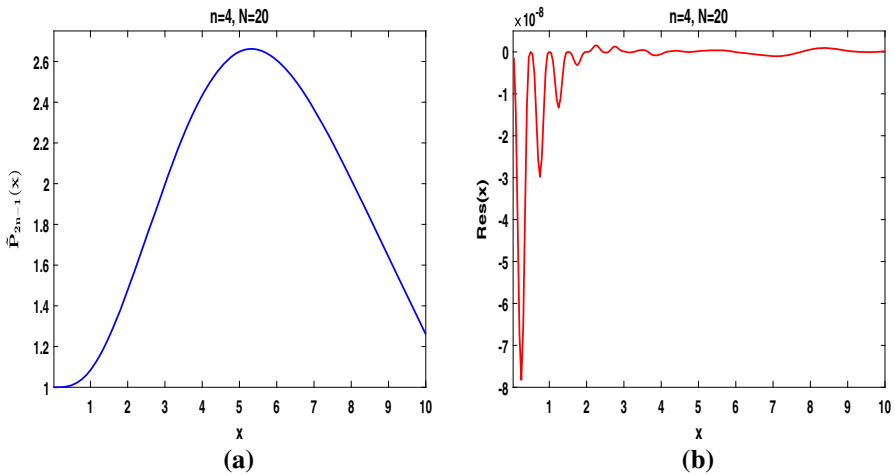


Fig. 9 **a** The approximate solution of Example 4.9 and **b** The residual error of the multistep TTF for Example 4.9

Example 4.9 As the last example, we investigate the nonlinear VIE [70,72,73]

$$u(x) = 1 + \int_0^x (x - t)^3(4 - x + t)e^{t-x} \frac{u^4(t)}{1 + 2u^2(t) + 2u^4(t)} dt, \quad t \in [0, 10] \tag{4.2}$$

which arises in the analysis of neural networks with post-inhibitory rebound [72]. The exact solution of this problem at $x = 10$ is $u(10) = 1.25995582337233$ [73]. We have utilized the multistep TTF with $n = 4$ and $N = 20$ to estimate the solution of this problem. The obtained estimation is presented in Fig. 9a. In order to study the accuracy of the multistep TTF, we plot in Fig. 9b the residual error based on VIE (4.2)

Table 13 Absolute errors of the multistep TTF with different values of n and N for Example 4.9 at $x = 10$

N	n			
	2	4	6	8
5	1.37e-02	5.57e-04	1.21e-06	3.57e-11
10	4.57e-04	3.04e-06	7.09e-10	1.35e-13
15	4.87e-05	1.30e-07	6.39e-12	2.29e-14
20	9.92e-06	1.36e-08	2.09e-13	2.14e-14

which is defined as

$$\text{Res}(x) = \mathcal{P}_{2n-1}^N(u; x) - 1 - \int_0^x (x-t)^3(4-x+t)e^{t-x} \frac{(\mathcal{P}_{2n-1}^N(u; t))^4}{1 + 2(\mathcal{P}_{2n-1}^N(u; t))^2 + 2(\mathcal{P}_{2n-1}^N(u; t))^4} dt.$$

The results were obtained in 18.829 s of computational time. We also display the influence of increasing n and N on the accuracy of the presented approximation in Table 13.

5 Conclusion

Integral equations are applicable tools in the modeling of many physical phenomena. Therefore, several numerical techniques have been applied to approximate the solution of them. In this article, we presented the TTF to solve linear and nonlinear VIEs of the second kind and systems of such equations. The proposed method can achieve smooth approximations of the solution of VIEs without the need of any collocation points. We investigated the convergence analysis of the presented method. The accuracy of the approximation can be controlled by the number of the basis functions. We also presented the multistep form of the TTF to increase the precision of the approximation over large intervals. This method reduces the solution of a VIE over a large interval into the solution of a set of VIEs over small subintervals. Moreover, This method utilizes in each step the computed results of the previous step. Subsequently, there is no need to increase the number of bases and find a large number of unknowns in each step. In order to increase the speed of the implemented algorithm, we estimated the integrals which appeared in the approximations by using the Gauss–Legendre quadrature formula. The obtained algorithm can provide accurate approximations to the solution of VIEs of the second kind in acceptable computational times. The method can compute the exact solution of the problems which have polynomial analytical solutions. It can also give accurate approximation to the solution of the other VIEs. The capability of the method confirmed by employing it to solve several illustrative examples. Future work will involve the development of this method for solving nonlinear integro-differential equations and nonstandard Volterra integral equations. Also, we intend to extend the proposed idea by employing the general form of the Hermite interpolation.

References

1. Baratella, P.: A Nyström interpolant for some weakly singular linear Volterra integral equations. *J. Comput. Appl. Math.* **231**, 725–734 (2009)
2. Bartoshevich, M.A.: A heat-conduction problem. *J. Eng. Phys.* **28**, 240–244 (1975)
3. Ding, H.J., Wang, H.M., Chen, W.Q.: Analytical solution for the electroelastic dynamics of a nonhomogeneous spherically isotropic piezoelectric hollow sphere. *Arch. Appl. Mech.* **73**, 49–62 (2003)
4. Farengo, R., Lee, Y.C., Guzdar, P.N.: An electromagnetic integral equation: application to microtearing modes. *Phys. Fluids* **26**, 3515–3523 (1983)
5. Wazwaz, A.: *Linear and Nonlinear Integral Equations*, vol. 639. Springer, Berlin (2011)
6. Rahman, M.: *Integral Equations and Their Applications*. WIT Press, Southampton (2007)
7. Ladopoulos, E.G.: *Singular Integral Equations: Linear and Non-linear Theory and Its Applications in Science and Engineering*. Springer, Berlin (2013)
8. Serov, V.S., Schürmann, H.W., Svetogorova, E.: Integral equation approach to reflection and transmission of a plane TE-wave at a (linear/nonlinear) dielectric film with spatially varying permittivity. *J. Phys. A Math. Gen.* **37**, 3489 (2004)
9. Yousefi, S.A.: Numerical solution of Abel's integral equation by using Legendre wavelets. *Appl. Math. Comput.* **175**, 574–580 (2006)
10. Hansen, P.C., Jensen, T.K.: *Large-Scale Methods in Image Deblurring*. International Workshop on Applied Parallel Computing. Springer, Berlin (2006)
11. Schürmann, H.W., Serov, V.S., Shestopalov, Y.V.: Reflection and transmission of a plane TE-wave at a lossless nonlinear dielectric film. *Phys. D* **158**, 197–215 (2001)
12. Hethcote, H.W., Lewis, M.A., Van Den Driessche, P.: An epidemiological model with a delay and a nonlinear incidence rate. *J. Math. Biol.* **27**, 49–64 (1989)
13. Van Den Driessche, P., Watmough, J.: A simple SIS epidemic model with a backward bifurcation. *J. Math. Biol.* **40**, 525–540 (2000)
14. Conte, D., Paternoster, B.: Multistep collocation methods for Volterra integral equations. *Appl. Numer. Math.* **59**, 1721–1736 (2009)
15. Cherrault, Y., Zitoun, F.B.: A Taylor expansion approach using Faà di Bruno's formula for solving nonlinear integral equations of the second and third kind. *Kybernetes* **38**, 19 (2009)
16. Maleknejad, K., Mollapourasl, R., Shahabi, M.: On the solution of a nonlinear integral equation on the basis of a fixed point technique and cubic B-spline scaling functions. *J. Comput. Appl. Math.* **239**, 346–358 (2013)
17. Maleknejad, K., Najafi, E.: Numerical solution of nonlinear Volterra integral equations using the idea of quasilinearization. *Commun. Nonlinear Sci. Numer. Simul.* **16**, 93–100 (2011)
18. Maleknejad, K., Rashidinia, J., Jalilian, H.: Non-polynomial spline functions and Quasi-linearization to approximate nonlinear Volterra integral equation. *Filomat* **32**, 3947–3956 (2018)
19. Mirzaee, F., Hoseini, S.F.: Hybrid functions of Bernstein polynomials and block-pulse functions for solving optimal control of the nonlinear Volterra integral equations. *Indagationes Mathematicae* **27**, 835–849 (2016)
20. Mirzaee, F., Hadadiyan, E.: Applying the modified block-pulse functions to solve the three-dimensional Volterra-Fredholm integral equations. *Appl. Math. Comput.* **265**, 759–767 (2015)
21. Wazwaz, A., Rach, R., Duan, J.S.: Adomian decomposition method for solving the Volterra integral form of the Lan-Emden equations with initial values and boundary conditions. *Appl. Math. Comput.* **219**, 5004–5019 (2013)
22. Vosughi, H., Shivanian, E., Abbasbandy, S.: A new analytical technique to solve Volterra's integral equations. *Math. Methods Appl. Sci.* **34**, 1243–1253 (2011)
23. Kant, K., Nelakanti, G.: Approximation methods for second kind weakly singular Volterra integral equations. *J. Comput. Appl. Math.* **368**, 112531 (2020)
24. Assari, P., Dehghan, M.: A meshless local Galerkin method for solving Volterra integral equations deduced from nonlinear fractional differential equations using the moving least squares technique. *Appl. Numer. Math.* **143**, 276–299 (2019)
25. Assari, P., Adibi, H., Dehghan, M.: A meshless discrete Galerkin (MDG) method for the numerical solution of integral equations with logarithmic kernels. *J. Comput. Appl. Math.* **267**, 160–181 (2014)
26. Shamooshaky, M.M., Assari, P., Adibi, H.: The numerical solution of nonlinear Fredholm-Hammerstein integral equations of the second kind utilizing Chebyshev wavelets. *J. Math. Comput. Sci.* **10**, 235–246 (2014)

27. Avazzadeh, Z., Heydari, M., Chen, W., Loghmani, G.B.: Smooth solution of partial integrodifferential equations using radial basis functions. *J. Appl. Anal. Comput.* **4**, 115–127 (2014)
28. Assari, P., Dehghan, M.: The approximate solution of nonlinear Volterra integral equations of the second kind using radial basis functions. *Appl. Numer. Math.* **131**, 140–157 (2018)
29. Assari, P., Adibi, H., Dehghan, M.: The numerical solution of weakly singular integral equations based on the meshless product integration (MPI) method with error analysis. *Appl. Numer. Math.* **81**, 76–93 (2014)
30. Assari, P., Adibi, H., Dehghan, M.: A meshless method for solving nonlinear two-dimensional integral equations of the second kind on non-rectangular domains using radial basis functions with error analysis. *J. Comput. Appl. Math.* **239**, 72–92 (2013)
31. Assari, P., Adibi, H., Dehghan, M.: A numerical method for solving linear integral equations of the second kind on the non-rectangular domains based on the meshless method. *Appl. Math. Model.* **37**, 9269–9294 (2013)
32. Mirzaee, F., Bimesl, S.: An efficient numerical approach for solving systems of high-order linear Volterra integral equations. *Sci. Iran.* **21**, 2250–2263 (2014)
33. Mirzaee, F., Hoseini, S.F.: Application of Fibonacci collocation method for solving Volterra-Fredholm integral equations. *Appl. Math. Comput.* **273**, 637–644 (2016)
34. Mirzaee, F., Bimesl, S.: A new Euler matrix method for solving systems of linear Volterra integral equations with variable coefficients. *J. Egypt. Math. Soc.* **22**, 238–248 (2014)
35. Mirzaee, F., Hadadiyan, E.: A new numerical method for solving two-dimensional Volterra-Fredholm integral equations. *J. Appl. Math. Comput.* **52**, 489–513 (2016)
36. Ahmadiania, M., Afshari, H.A., Heydari, M.: Numerical solution of Itô-Volterra integral equation by least squares method. *Numer. Algorithms* **84**, 591–602 (2020)
37. Assari, P., Adibi, H., Dehghan, M.: A meshless method based on the moving least squares (MLS) approximation for the numerical solution of two-dimensional nonlinear integral equations of the second kind on non-rectangular domains. *Numer. Algorithms* **67**, 423–455 (2014)
38. Saberirad, F., Karbassi, S.M., Heydari, M.: Numerical solution of nonlinear fuzzy Volterra integral equations of the second kind for changing sign kernels. *Soft. Comput.* **23**, 11181–11197 (2019)
39. De Hoog, F., Weiss, R.: Implicit Runge-Kutta methods for second kind Volterra integral equations. *Numer. Math.* **23**, 199–213 (1974)
40. Jiang, W., Chen, Z.: Solving a system of linear Volterra integral equations using the new reproducing kernel method. *Appl. Math. Comput.* **219**, 10225–10230 (2013)
41. Saffarzadeh, M., Loghmani, G.B., Heydari, M.: An iterative technique for the numerical solution of nonlinear stochastic Itô-Volterra integral equations. *J. Comput. Appl. Math.* **333**, 74–86 (2018)
42. Saffarzadeh, M., Heydari, M., Loghmani, G.B.: Convergence analysis of an iterative numerical algorithm for solving nonlinear stochastic Itô-Volterra integral equations with m-dimensional Brownian motion. *Appl. Numer. Math.* **146**, 182–198 (2019)
43. Saffarzadeh, M., Heydari, M., Loghmani, G.B.: Convergence analysis of an iterative algorithm to solve system of nonlinear stochastic Itô-Volterra integral equations. *Math. Methods Appl. Sci.* **43**, 5212–5233 (2020)
44. Eshkuvatov, K., Hameed, H.H., Taib, B.M., Nik Longcd, N.M.A.: General 2×2 system of nonlinear integral equations and its approximate solution. *J. Comput. Appl. Math.* **361**, 528–546 (2019)
45. Babolian, E., Mordad, M.: A numerical method for solving systems of linear and nonlinear integral equations of the second kind by hat basis functions. *Comput. Math. Appl.* **62**, 187–198 (2011)
46. Mirzaee, F., Hadadiyan, E.: Numerical solution of Volterra-Fredholm integral equations via modification of hat functions. *Appl. Math. Comput.* **280**, 110–123 (2016)
47. Mirzaee, F., Hadadiyan, E.: Numerical solution of optimal control problem of the non-linear Volterra integral equations via generalized hat functions. *IMA J. Math. Control Inform.* **34**, 889–904 (2017)
48. Mirzaee, F., Hadadiyan, E.: Using operational matrix for solving nonlinear class of mixed Volterra-Fredholm integral equations. *Math. Methods Appl. Sci.* **40**, 3433–3444 (2017)
49. Mirzaee, F., Hadadiyan, E.: Application of two-dimensional hat functions for solving space-time integral equations. *J. Appl. Math. Comput.* **51**, 453–486 (2016)
50. Heydari, M., Avazzadeh, Z., Loghmani, G.B.: Chebyshev cardinal functions for solving Volterra-Fredholm integrodifferential equations using operational matrices. *Iran. J. Sci. Technol. (Sci.)* **36**, 13–24 (2012)
51. Brunner, H.: *Volterra Integral Equations: An Introduction to Theory and Applications*, vol. 30. Cambridge University Press, Cambridge (2017)

52. Günerhan, H., Khodadad, F.S., Rezazadeh, H., Khater, M.M.: Exact optical solutions of the $(2 + 1)$ dimensions Kundu-Mukherjee-Naskar model via the new extended direct algebraic method. *Mod. Phys. Lett. B* **2050225**, 2050225 (2020)
53. Attia, R.A., Lu, D., Ak, T., Khater, M.M.: Optical wave solutions of the higher-order nonlinear Schrödinger equation with the non-Kerr nonlinear term via modified Khater method. *Mod. Phys. Lett. B* **34**, 2050044 (2020)
54. Ali, A.T., Khater, M.M., Attia, R.A., Abdel-aty, A.H., Lu, D.: Abundant numerical and analytical solutions of the generalized formula of Hirota-Satsuma coupled KdV system. *Chaos Solitons Fract.* **131**, 109473 (2020)
55. Khater, M.M., Attia, R.A., Abdel-Aty, A.H., Abdou, M.A., Eleuch, H., Lu, D.: Analytical and semi-analytical ample solutions of the higher-order nonlinear Schrödinger equation with the non-Kerr nonlinear term. *Res. Phys.* **16**, 103000 (2020)
56. Park, C., Khater, M.M., Attia, R.A., Alharbi, W., Alodhaibi, S.S.: An explicit plethora of solution for the fractional nonlinear model of the low-pass electrical transmission lines via Atangana-Baleanu derivative operator. *Alexandr. Eng. J.* **59**, 1205–1214 (2020)
57. Park, C., Khater, M.M., Abdel-Aty, A.H., Attia, R.A., Lu, D.: On new computational and numerical solutions of the modified Zakharov-Kuznetsov equation arising in electrical engineering. *Alexandr. Eng. J.* **59**, 1099–1105 (2020)
58. Phillips, G.M.: Explicit forms for certain Hermite approximations. *BIT Numer. Math.* **13**, 177–180 (1973)
59. Costabile, F.A., Napoli, A.: Solving BVPs using two-point Taylor formula by a symbolic software. *J. Comput. Appl. Math.* **210**, 136–148 (2007)
60. Costabile, F.A., Napoli, A.: Collocation for high order differential equations with two-points Hermite boundary conditions. *Appl. Numer. Math.* **87**, 157–167 (2015)
61. Karamollahi, N., Loghmani, G.B., Heydari, M.: Dual solutions of the nonlinear problem of heat transfer in a straight fin with temperature-dependent heat transfer coefficient. *Int. J. Numer. Methods Heat Fluid Flow* (2020). <https://doi.org/10.1108/HFF-04-2020-0201>
62. Sugiyama, S.: Stability problems on difference and functional-differential equations. *Proc. Jpn. Acad.* **45**, 526–529 (1969)
63. Granas, A., Dugundji, J.: *Fixed Point Theory*. Springer, Berlin (2013)
64. Fazeli, S., Somayyeh, G., Hojjati, S.: Shahmorad: Super implicit multistep collocation methods for nonlinear Volterra integral equations. *Math. Comput. Model.* **55**, 590–607 (2012)
65. Erfanian, M.: The approximate solution of nonlinear integral equations with the RH wavelet bases in a complex plane. *Int. J. Appl. Comput. Math.* **4**, 31 (2018)
66. Saberi-Nadjafi, J., Mehrabinezhad, M., Diogo, T.: The Coifflet-Galerkin method for linear Volterra integral equations. *Appl. Math. Comput.* **221**, 469–483 (2013)
67. Berenguer, M.I., et al.: Biorthogonal systems for solving Volterra integral equation systems of the second kind. *J. Comput. Appl. Math.* **235**, 1875–1883 (2011)
68. Katani, R., Shahmorad, S.: Block by block method for the systems of nonlinear Volterra integral equations. *Appl. Math. Model.* **34**, 400–406 (2010)
69. Sekar, Chandra Guru, R., Murugesan, K. : STWS approach for Hammerstein system of nonlinear Volterra integral equations of the second kind. *Int. J. Comput. Math.* **94**, 1867–1878 (2017)
70. Abdi, A., Hojjati, G., Jackiewicz, Z., Mahdi, H.: A new code for Volterra integral equations based on natural Runge-Kutta methods. *Appl. Numer. Math.* **143**, 35–50 (2019)
71. Abdi, A., Hosseini, S.A., Podhaisky, H.: Numerical methods based on the Floater-Hormann interpolants for stiff VIEs. *Numer. Algorithms* **85**, 867–886 (2019)
72. Hairer, E., Lubich, C., Schlichte, M.: Fast numerical solution of nonlinear Volterra convolution equations. *SIAM J. Sci. Stat. Comput.* **6**, 532–541 (1985)
73. Capobianco, G., Conte, D., Del Prete, I., Russo, E.: Fast Runge-Kutta methods for nonlinear convolution systems of Volterra integral equations. *BIT Numer. Math.* **47**, 259–275 (2007)