

Research Article

On the Multiwavelets Galerkin Solution of the Volterra–Fredholm Integral Equations by an Efficient Algorithm

H. Bin Jebreen 

Department of Mathematics, College of Science, King Saud University, P.O. Box 2455, Riyadh 11451, Saudi Arabia

Correspondence should be addressed to H. Bin Jebreen; hjebreen@ksu.edu.sa

Received 18 July 2020; Revised 19 October 2020; Accepted 24 October 2020; Published 11 November 2020

Academic Editor: Shiv Kaushik

Copyright © 2020 H. Bin Jebreen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We develop the multiwavelet Galerkin method to solve the Volterra–Fredholm integral equations. To this end, we represent the Volterra and Fredholm operators in multiwavelet bases. Then, we reduce the problem to a linear or nonlinear system of algebraic equations. The interesting results arise in the linear type where thresholding is employed to decrease the nonzero entries of the coefficient matrix, and thus, this leads to reduction in computational efforts. The convergence analysis is investigated, and numerical experiments guarantee it. To show the applicability of the method, we compare it with other methods and it can be shown that our results are better than others.

1. Introduction

A mathematical model of the spatiotemporal development of an epidemic yields the following Volterra–Fredholm integral equation (VFIE):

$$u(x) = f(x) + \int_0^x k_1(x, s, u(s)) ds + \int_0^1 k_2(x, s, u(s)) ds, \quad x \in \Omega := [0, 1], \quad (1)$$

where the given functions $f: \Omega \rightarrow \mathbb{R}$ and $k_1: S \times \mathbb{R} \rightarrow \mathbb{R}$ with $S := \{(x, s): x, s \in \Omega\}$ are assumed to be continuous functions. Furthermore, we consider $k_2 := p(x, s)h(u(s))$ to be integrable, where $h(x)$ is a nonlinear function and $p(x, s): S \rightarrow \mathbb{R}$ is a continuous function. Besides, the given functions are selected so that equation (1) has a unique solution.

The parabolic boundary value problems lead to these types of integral equations and widely arise from various physical and biological models. The VFIE also appears in the literature in mixed form as

$$u(x) = f(x) + \int_0^x \int_0^1 k(t, s)u(s) ds dt, \quad (2)$$

where f and k are analytic functions. Many authors studied the mixed form of VFIE numerically. Among these, we can mention collocation method [1], projection method [2], spline collocation method [3], wavelet collocation method [4], Adomian decomposition method [5], and so on [6–8]. Among these studies, we focus on a paper that uses the multiwavelet Galerkin method to solve linear mixed VFIE as mentioned in [9]. In [9], the wavelet transform matrix and the operational matrix of integration are utilized to reduce the problem of linear mixed VFIE to a sparse linear system of algebraic equations. By searching among the sources, we can find a small number of papers in the field of numerical and analytical solutions to problem (1). In [10], the Lagrange collocation method is employed to solve this problem. Wang and Wang [11] applied the Taylor collocation method to find the numerical solution of the equation. Also, the convergence analysis is investigated for the proposed method. The Taylor collocation method was applied by Karamete and Sezer [12] to solve this equation as well as the high-order linear Fredholm–Volterra integrodifferential equations [13]. The Bell polynomials have been employed for solving this equation [14].

The motivation of our work is to develop the multiwavelet Galerkin method to solve (1). We split the problem into two configurations, linear and nonlinear. After using

the multiwavelet Galerkin method, both types reduce to the system of the linear and nonlinear algebraic equations, respectively. The interesting results arise in the linear type where thresholding is employed to decrease the nonzero entries of the coefficient matrix. This gives the sparse system. This property is very useful to reduce the computational cost. We use Alpert's multiwavelet bases constructed in [15] following [16], and these bases have been used to solved PDEs, ODEs, and integral equations [9, 17–19]. These bases allow us to have high vanishing moments, compact support, and properties such as orthogonality and interpolation [15]. These characteristics of Alpert's multiwavelets lead to a sparse representation of differential and integral operators [16, 20].

This paper is organized as follows: In Section 2, we briefly introduce Alpert's multiwavelet bases. In Section 3, the Multiwavelets Galerkin method is used to solve this problem, and the conditions for convergence of the proposed method are discussed. Section 5 contains some numerical results to confirm the validity and efficiency of the proposed method, and Section 6 contains a few concluding remarks.

2. Alpert's Multiwavelet Bases

Let $J \in \mathbb{Z}^+ \cup \{0\}$. We consider the uniform finite discretizations $\Omega := [0, 1] = \cup_{b \in \mathcal{B}} X_{J,b}$ where the subintervals $X_{J,b} := [x_b, x_{b+1}]$ are determined by the point $x_b := (b/(2^J))$ with $\mathcal{B} := \{0, \dots, 2^J - 1\}$. For $k \in \mathcal{R} := \{0, 1, \dots, r - 1\}$, we introduce the subspace V_J^r as a space of piecewise polynomial bases of degree less than multiplicity parameter r that is spanned by

$$V_J^r := \text{span}\{\phi_{j,b}^k := \mathcal{D}_{2^j} \mathcal{T}_b \phi^k, b \in \mathcal{B}_j, k \in \mathcal{R}\} \subset L^2(\Omega), \quad r \geq 0, \quad (3)$$

where \mathcal{D} and \mathcal{T} are the dilation and translation operators, respectively, and $\{\phi^k\}_{k \in \mathcal{R}}$ are the primal interpolating scaling bases introduced by Alpert et al. [16]. Given nodes $\{\tau_k\}_{k \in \mathcal{R}}$, which are the roots of Legendre polynomial of degree r , the interpolating scaling bases are defined as

$$\phi^k(t) = \begin{cases} \sqrt{\frac{2}{\omega_k}} L_k(2t - 1), & t \in [0, 1], \\ 0, & \text{otherwise,} \end{cases} \quad (4)$$

where $\{L_k(t)\}_{k \in \mathcal{R}}$ are the Lagrange interpolating polynomials at the point $\{\tau_k\}_{k \in \mathcal{R}}$ and $\{\omega_k\}_{k \in \mathcal{R}}$ are the Gauss–Legendre quadrature weights [16, 18]. These bases form an orthonormal bases on Ω with respect to the L^2 -inner product. Due to the definition of the space V_J^r , the spaces $\{V_j^r\}_{j \in \mathbb{Z}^+ \cup \{0\}}$ have dimension $N := 2^j r$ and obviously are nested:

$$V_0^r \subset V_1^r \subset \dots \subset V_J^r \subset \dots \subset L^2(\Omega). \quad (5)$$

Hence, we consider the complement subspace W_J^r of V_J^r in V_{J+1}^r such that

$$V_{J+1}^r = V_J^r \oplus W_J^r, W_J^r \perp V_J^r, \quad (6)$$

where \oplus denotes orthogonal sums. According to (6), the space V_J may be inductively decomposed to

$$V_J^r = V_0^r \oplus \left(\oplus_{j=0}^{J-1} W_j^r\right). \quad (7)$$

The complement subspace W_J^r has dimension $2^J r$ and is spanned by multiwavelet bases $\{\psi^k\}_{k \in \mathcal{R}}$, as

$$W_J^r = \text{span}\{\psi_{J,b}^k \cong W_{2^j} W_b \psi^k: b \in \mathcal{B}_J, k \in \mathcal{R}\}. \quad (8)$$

Because Alpert's multiwavelets are completely introduced in [16], we avoid this and refer the readers to [15, 16, 19].

Every function $p \in L^2(\Omega)$ can be represented in the form

$$p \approx V_J^r(p) = \sum_{b \in \mathcal{B}_J} \sum_{k \in \mathcal{R}} p_{J,b}^k \phi_{J,b}^k, \quad (9)$$

where \mathcal{P}_J^r is the orthogonal projection that maps $L^2(\Omega)$ onto the subspace V_J^r . To find the coefficients $p_{J,b}^k$ that are determined by $\langle p, \phi_{J,b}^k \rangle = \int_{X_{J,b}} f(x) \phi_{J,b}^k(x) dx$, we shall compute the integrals. We apply the r -point Gauss–Legendre quadrature by a suitable choice of the weights ω_k and nodes τ_k for $k \in \mathcal{R}$ to avoid these integrals [16, 19], via

$$p_{J,b}^k \approx 2^{-(J/2)} \sqrt{\frac{\omega_k}{2}} p\left(2^{-J} \left(\frac{\tau_k + 1}{2} + b\right)\right), \quad k \in \mathcal{R}, b \in \mathcal{B}_J. \quad (10)$$

Convergence analysis of the projection $\mathcal{P}_J^r(p)$ is investigated for the r -times continuously on differentiable function $p \in C^r(\Omega)$:

$$\|\mathcal{P}_J^r(p) - p\| \leq 2^{-Jr} \frac{2}{4^r r!} \sup_{x \in [0,1]} |p^{(r)}(x)|. \quad (11)$$

For the full proof of this approximation and further details, we refer the readers to [15]. Thus, we can conclude that $\mathcal{P}_J^r(p)$ converges to p with rate of convergence $O(2^{-Jr})$.

Assume that the vector function $\Phi_J^r := [\Phi_{r,J,0}, \dots, \Phi_{r,J,2^J-1}]^T$ with $\Phi_{r,J,b} := [\phi_{J,b}^0, \dots, \phi_{J,b}^{r-1}]$ includes the scaling functions and is called multiscaling function. Approximation (9) may be rewritten using the vector P that includes entries $P_{br+k+1} := p_{J,b}^k$ as follows:

$$\Phi_J^r(p) = P^T \Phi_J^r, \quad (12)$$

where P is an N -dimensional vector. The building blocks of these bases construction can be applied to approximate a higher-dimensional function. To this end, one can introduce the two-dimensional subspace $V_J^{r,2} := V_J^r \times V_J^r \subset L^2(\Omega)^2$ that is spanned by

$$\{\phi_{J,b}^k \phi_{J,b'}^{k'}: b, b' \in \mathcal{B}_J, k, k' \in \mathcal{R}\}. \quad (13)$$

Thus, by this assumption, to derive an approximation of the function $p \in L^2(\Omega)^2$ by the projection operator \mathcal{P}_J^r , we have

$$p(x, y) \approx \Phi_J^r(p)(x, y) = \Phi_J^{rT}(x) P \Phi_J^r(y), \quad (14)$$

where the components of the square matrix P of order N are obtained by

$$\mathcal{P}_{rb+(k+1),rb'+(k'+1)} \approx 2^{-J} \sqrt{\frac{\omega_k}{2}} \sqrt{\frac{\omega_{k'}}{2}} p(2^{-J}(\hat{\tau}_k + b), 2^{-J}(\hat{\tau}_{k'} + b')), \quad (15)$$

where $\hat{\tau}_k = (\tau_k + 1)/2$. Consider the $2r$ -th partial derivatives of $p: \Omega^2 \rightarrow \mathbb{R}$ to be continuous. Utilizing this assumption, the error of this approximation can be bounded as follows:

$$\|\mathcal{P}_J^r p - p\| \leq \mathcal{M}_{\max} \frac{2^{1-rJ}}{4^r r!} \left(2 + \frac{2^{1-Jr}}{4^r r!} \right), \quad (16)$$

where \mathcal{M}_{\max} is a constant.

Given orthogonal projection operator $\mathcal{Q}_j^r = \mathcal{P}_{j+1}^r - \mathcal{P}_j^r$ that maps $L^2(\Omega)$ onto W_j^r , the multiscale projection operator \mathcal{M}_J^r can be represented as

$$\mathcal{M}_J^r = \mathcal{Q}_0^r + \sum_{j=0}^{J-1} \mathcal{Q}_j^r, \quad (17)$$

and consequently, any function $p \in L^2(\Omega)$ can be approximated as a linear combination of multiwavelets and single-scale interpolating scaling functions as

$$p \approx \mathcal{M}_J^r(p) = \sum_{k=0}^{r-1} p_{0,0}^k \phi_{0,0}^k + \sum_{j=0}^{J-1} \sum_{b \in \mathcal{B}_j} \sum_{k \in \mathcal{R}} \tilde{p}_{j,b}^k \psi_{j,b}^k, \quad (18)$$

where

$$\begin{aligned} p_{0,0}^k &\equiv \langle p, \phi_{0,0}^k \rangle, \\ \tilde{p}_{j,b}^k &\equiv \langle p, \psi_{j,b}^k \rangle. \end{aligned} \quad (19)$$

Note that we can compute the coefficients $p_{0,0}^k$ by using (10). But in many cases, multiwavelet coefficients from zero up to higher-level $J - 1$ must be evaluated numerically. To avoid this, we use multiwavelet transform matrix T_J , introduced in [18, 19]. This matrix connects multiwavelet bases and multiscaling functions, via

$$\Psi_J^r = T_J \Phi_J^r, \quad (20)$$

where $\Psi_J^r = [\Phi_{r,0,b}, \Psi_{r,0,b}, \Psi_{r,1,b}, \dots, \Psi_{r,J-1,b}]^T$ is a vector with the same dimension Φ_J^r (here $\Psi_{r,j,b} = [\psi_{j,b}^0, \dots, \psi_{j,b}^{r-1}]$). This representation helps to rewrite equation (18) as to form

$$p \approx \mathcal{M}_J^r(p) = \tilde{P}_J^T \Psi_J^r, \quad (21)$$

where we have the N -dimensional vector \tilde{P}_J whose entries are $p_{0,0}^k$ and $\tilde{p}_{j,b}^k$ and is given by employing the multiwavelet transform matrix T_J as $\tilde{P}_J = T_J P_J$.

The multiwavelet coefficients (details) become small when the underlying function is smooth (locally) with increasing refinement levels. If the multiwavelet bases have N_ψ^r vanishing moments [19, 21], then details decay at the rate of $2^{-JN_\psi^r}$ [17]. Because vanishing moments of Alpert's multiwavelets are equal to r , consequently, one can obtain $\tilde{p}_{j,b}^k \approx O(2^{-Jr})$. This allows us to truncate the full wavelet transforms while preserving most of the necessary data.

Thus, we can set to zero all details that satisfy a certain constraint ε using thresholding operator \mathcal{C}_ε :

$$\mathcal{C}_\varepsilon(\tilde{P}_J) = \bar{P}_J, \quad (22)$$

and the elements of \bar{P}_J are determined by

$$\bar{p}_{j,b}^k \equiv \begin{cases} \tilde{p}_{j,b}^k, & (j, b, k) \in D_\varepsilon, \\ 0, & \text{else,} \end{cases} \quad b \in \mathcal{B}_j, j = 0, \dots, J-1, k \in \mathcal{R}, \quad (23)$$

where $D_\varepsilon = \{(j, b, k): |\tilde{p}_{j,b}^k| > \varepsilon\}$. Now, we can bound the approximation error after thresholding [17] via

$$\|\mathcal{P}_J^r p - \mathcal{P}_{J,D_\varepsilon}^r p\|_{L^2(\Omega)} \leq C_{\text{thr}} \varepsilon, \quad (24)$$

where $\mathcal{P}_{J,D_\varepsilon}^r(p)$ is the projection operator after thresholding with the threshold ε and $C_{\text{thr}} > 0$ is a constant independent of J and ε .

3. Multiwavelet Galerkin Method

In order to obtain multiwavelet Galerkin solution of (1), assume that solution can be approximated as an expansion of the Alpert's multiwavelets, i.e.,

$$u(x) \approx \mathcal{P}_J^r(u)(x) = U^T \Psi_J^r(x), \quad (25)$$

where the N dimension vector U of unknowns must be specified. This solution is selected such that it satisfies (1) approximately. Also, it is obtained from the solution of the minimization problem

$$\|u - \mathcal{P}_J^r(u)\| = \min_{z \in L^2(\Omega)} \|u - z\|. \quad (26)$$

Since $L^2(\Omega)$ is an inner product space with finite dimension, it can be shown that this minimization problem has a unique solution [22].

Let us rewrite (1) in the operator form

$$(I - \mathcal{F})u = g, \quad (27)$$

where $g(x) = f(x) + V(u)(x)$ with $\mathcal{V}(u)(x) = \int_0^x k_1(x, s, u(s)) ds$. Furthermore, the operator $\mathcal{F}(u) = \int_0^x P(x, s) h(u(s)) ds$ is assumed to be compact on $L^2(\Omega)$ to $L^2(\Omega)$ and k_1 is a given continuous function. Due to these assumptions, $V(u)(x)$ is a continuous function and, consequently, g is also continuous function. Due to (11), the function $g(x)$ can be approximated at a rate of at least 2^{-Jr} :

$$\begin{aligned} g(x) &\approx \mathcal{P}_J^r(g)(x) = F^T \Phi_J^r(x) + K_1^T \Phi_J^r(x) \\ &= F^T T_J^{-1} \Psi_J^r(x) + K_1^T T_J^{-1} \Psi_J^r(x) \\ &= \tilde{F}^T \Psi_J^r(x) + \begin{cases} \mathcal{A}_1^T \Psi_J^r(x), & \text{linear,} \\ \mathcal{C}_1 \Psi_J^r(x), & \text{nonlinear,} \end{cases} \\ &= \tilde{G}^T \Psi_J^r(x) = \mathcal{M}_J^r(x)(x), \end{aligned} \quad (28)$$

where \mathcal{A}_1 and \mathcal{C}_1 are $N \times N$ matrices and the rest are N -dimensional vectors. Now, we introduce the residual in the approximation of (1):

$$r_J(x) = u_J(x) - \mathcal{M}_J^r(g)(x) - \mathcal{M}_J^r(\mathcal{F}_J^r)(x), \quad (29)$$

where $u_J := \mathcal{M}_J^r(u)$ and

$$\begin{aligned} \mathcal{M}_J^r \mathcal{F}_J^r(u_J) &= \mathcal{M}_J^r \left(\int_0^1 \mathcal{M}_J^r(k_2(x, s, u_J(s))) ds \right) \\ &= \mathcal{M}_J^r \left(\int_0^1 \mathcal{M}_J^r(p(x, s)h(u_J(s))) ds \right) \\ &= \int_0^1 H^T \Psi_J^r(s) \Psi_J^{rT}(s) P \Psi_J^r(x) ds = H^T P \Psi_J^r(x) \\ &= \begin{cases} U^T \Psi_2 \Psi_J^r(x), & \text{linear,} \\ \Psi_2 \Psi_J^r(x), & \text{nonlinear.} \end{cases} \end{aligned} \quad (30)$$

Symbolically,

$$r_J = \mathcal{M}_J^r((I - \mathcal{F}_J^r)u_J - g). \quad (31)$$

To find u_J , it requires that the approximate solution u_J satisfies

$$\langle r_J, [\Psi_J^r]_i \rangle = 0, \quad i = 1, \dots, N. \quad (32)$$

This is multiwavelet Galerkin's method and yields a linear or nonlinear system that must be solved to obtain the approximate solution. Note that $\mathcal{M}_J^r(z) = 0$ if and only if $\langle z, [\Psi_J^r]_i \rangle = 0$. Thus, we can rewrite (31) as

$$\mathcal{M}_J^r(r_J) = 0, \quad (33)$$

or equivalently,

$$\mathcal{M}_J^r(I - \mathcal{F}_J^r)u_J = \mathcal{M}_J^r(g). \quad (34)$$

Note that $\mathcal{M}_J^r(u_J) = u_J$ whenever $u_J \in V_J^r$. Due to this, equation (34) can be rewritten as

$$(I - \mathcal{M}_J^r \mathcal{F}_J^r)(u_J) = \mathcal{M}_J^r(g). \quad (35)$$

According to (35), we obtain the system of linear or nonlinear equations. Due to the higher vanishing moments r and increasing refinement level J , for the linear type of this equation, we can discard coefficients by hard thresholding introduced in the previous section. We can reduce the computational efforts using proper methods for this type of system such as the GMRES method. The GMRES method is introduced by Saad and Shultz [23] for solving sparse and large linear systems. The GMRES generates an approximate solution whose residual norm is minimum by using a Krylov subspace. In this paper, we use restarted GMRES Algorithm 2 [24]. To use this method, we must first define Arnoldi's algorithm. Arnoldi's procedure is an algorithm for building an orthogonal basis of the Krylov subspace κ_m . The N -th Krylov subspace is defined as follows:

$$\kappa_m(\Lambda, w) = \text{span}\{w_1, \Lambda w_1, \dots, \Lambda^{m-1} w_1\}. \quad (36)$$

Here, we assume that system (35) of the linear type to be of form $\Lambda U = D$ and W_m is a $N \times m$ matrix with column vectors w_1, \dots, w_m . Also, \bar{H}_m is a $(m+1 \times m)$ Hessenberg

matrix whose nonzero entries $h_{i,j}$ are defined by Algorithm 1.

- (1) Compute $r_0 = D - \Lambda U_0$, $\beta = \|r_0\|_2$ and $w_1 = r_0/\beta$
- (2) Generate the Arnoldi basis and the matrix \bar{H}_m using the Arnoldi algorithm starting with w_1
- (3) Compute y_m the minimizer of $\|\beta e_1 - \bar{H}_m y\|_2$ and $U_m = U_0 + W_m y_m$
- (4) If satisfied then stop, else set $U_0 = U_m$ and go to 1

To investigate the convergence analysis, one can prove that $\|\mathcal{F}_J^r - \mathcal{M}_J^r \mathcal{F}_J^r\| = O(2^{-Jr})$. Thus, $\|\mathcal{F}_J^r - \mathcal{M}_J^r \mathcal{F}_J^r\| \rightarrow 0$ when $J \rightarrow \infty$ because \mathcal{F} is a compact operator. Now, we can raise the convergence theorem.

Theorem 1. *Let \mathcal{F} be a compact operator and $I - \mathcal{F}$ be injective. Assume that the sequence $\mathcal{F}_J^r: L^2(\Omega) \rightarrow L^2(\Omega)$ is collectively compact and pointwise convergent to \mathcal{F} .*

Then, $(I - \mathcal{F}_J^r)^{-1}$ exists and is uniformly bounded. Also, the solution of (27) and (31) satisfy the error estimate

$$\|u - u_J\| \leq \frac{\|(I - \mathcal{F}_J^r)^{-1}\|}{1 - \varepsilon_{J_0} \|(I - \mathcal{F}_J^r)^{-1}\|} \left(\|u - \mathcal{M}_J^r u\| + \|\mathcal{M}_J^r \mathcal{F} - \mathcal{M}_J^r \mathcal{F}_J^r\| \|u\| \right). \quad (37)$$

Proof. Because the sequence \mathcal{F}_J^r converges pointwise to \mathcal{F} in $L^2(\Omega)$ and is collectively compact, we conclude that

$$\|(\mathcal{F}_J^r - \mathcal{F})\mathcal{F}_J^r\| \rightarrow 0, \quad \text{as } J \rightarrow \infty. \quad (38)$$

For all sufficiently large J , we have

$$\|(I - \mathcal{F})^{-1}(\mathcal{F}_J^r - \mathcal{F})\mathcal{F}_J^r\| < 1, \quad (39)$$

and as a consequence of this, $I - \mathcal{F}_J^r$ is reversible. Note that the inverse operator $(I - \mathcal{F})$ exists due to the Riesz theorem. The investigation is based on the approximation of $I - \mathcal{F}_J^r$ by $I - \mathcal{M}_J^r \mathcal{F}_J^r$,

$$\begin{aligned} I - \mathcal{M}_J^r \mathcal{F}_J^r &= (I - \mathcal{F}_J^r) + (\mathcal{F}_J^r - \mathcal{M}_J^r \mathcal{F}_J^r) \\ &= (I - \mathcal{F}_J^r) \left(I + (I - \mathcal{F}_J^r)^{-1} (\mathcal{F}_J^r - \mathcal{M}_J^r \mathcal{F}_J^r) \right). \end{aligned} \quad (40)$$

To prove the existence of $(I - \mathcal{M}_J^r \mathcal{F}_J^r)^{-1}$, assume that

$$\varepsilon_{J_0} \equiv \sup_{J \geq J_0} \|\mathcal{F}_J^r - \mathcal{M}_J^r \mathcal{F}_J^r\| < \frac{1}{\|(I - \mathcal{F}_J^r)^{-1}\|}. \quad (41)$$

Thus, $(I + (I - \mathcal{F}_J^r)^{-1} (\mathcal{F}_J^r - \mathcal{M}_J^r \mathcal{F}_J^r))^{-1}$ exists and is uniformly bounded due to geometric series theorem, i.e.,

$$\left\| \left(I + (I - \mathcal{F}_J^r)^{-1} (\mathcal{F}_J^r - \mathcal{M}_J^r \mathcal{F}_J^r) \right)^{-1} \right\| \leq \frac{1}{1 - \varepsilon_{J_0} \|(I - \mathcal{F}_J^r)^{-1}\|}. \quad (42)$$

Using (42), $(I - \mathcal{M}_J^r \mathcal{F}_J^r)$ exists. Taking norm and using (42), we obtain

$$\|(I - \mathcal{M}_J^r \mathcal{F}_J^r)^{-1}\| \leq \frac{\|(I - \mathcal{F}_J^r)^{-1}\|}{1 - \varepsilon_{J_0} \|(I - \mathcal{F}_J^r)^{-1}\|}. \quad (43)$$

Applying the operator \mathcal{M}_J^r to both sides of (27) and then rearranging, we can obtain

$$(I - \mathcal{M}_J^r \mathcal{F}_J^r)u = \mathcal{M}_J^r g + (u - \mathcal{M}_J^r u) + (\mathcal{M}_J^r \mathcal{F} - \mathcal{M}_J^r \mathcal{F}_J^r)u. \quad (44)$$

Subtracting (35) from (44), we obtain

$$(I - \mathcal{M}_J^r \mathcal{F}_J^r)(u - u_J) = (u - \mathcal{M}_J^r u) + (\mathcal{M}_J^r \mathcal{F} - \mathcal{M}_J^r \mathcal{F}_J^r)u. \quad (45)$$

Taking norm and employing (43),

$$\|u - u_J\| \leq \frac{\|(I - \mathcal{F}_J^r)^{-1}\|}{1 - \varepsilon_{J_0} \|(I - \mathcal{F}_J^r)^{-1}\|} (\|u - \mathcal{M}_J^r u\| + \|(\mathcal{M}_J^r \mathcal{F} - \mathcal{M}_J^r \mathcal{F}_J^r)u\|). \quad (46)$$

This is equivalent to (37). It is straightforward to show that $\|(\mathcal{M}_J^r \mathcal{F} - \mathcal{M}_J^r \mathcal{F}_J^r)u\| \rightarrow 0$ as $J \rightarrow \infty$. Assume that $\{u_J\}$ be a sequence of continuous functions so that $u_J \rightarrow u$ as $J \rightarrow \infty$. Since the orthonormal projection \mathcal{M}_J^r satisfies $\|\mathcal{M}_J^r\| = 1$, we can obtain

$$\begin{aligned} \|u - \mathcal{M}_J^r u\| &\leq \|u - u_J\| + \|u_J - \mathcal{M}_J^r u_J\| + \|\mathcal{M}_J^r(u - u_J)\| \\ &\leq 2\|u - u_J\| + \|u_J^r - \mathcal{M}_J^r u_J^r\|. \end{aligned} \quad (47)$$

Thus, for each real number $\varepsilon > 0$, there exists a number J_0 such that, for every number $J_0 \leq J$, one can write $\|u - u_J\| \leq (\varepsilon/4)$. This then implies that

$$\|u - \mathcal{M}_J^r u\| \leq \frac{\varepsilon}{2} + \|u_J^r - \mathcal{M}_J^r u_J^r\|. \quad (48)$$

This implies that $\|u - \mathcal{M}_J^r u\| \leq \varepsilon$, for sufficiently large value of J and because ε is arbitrary, consequently, $\mathcal{M}_J^r u \rightarrow u$ as $J \rightarrow \infty$. \square

4. Numerical Experiments

To verify the accuracy and efficiency of the proposed method, we consider a series of numerical examples. In the linear type of equation (1), we aim to generate a sparse matrix to reduce the computational costs. We illustrate the rate of sparsity S_ε which is defined by [21,25]

$$S_\varepsilon = \frac{N_0 - N_\varepsilon}{N_0} \times 100\%, \quad (49)$$

where ε is the threshold (small positive number) and N_ε and N_0 are the number of elements remaining after thresholding and the total number of elements, respectively.

Example 1. Consider the linear VFIE given in [11] as

$$u(x) = e^{-x} - e^x(x-1) + \int_0^x e^{x+s} u(s) ds - \int_0^1 e^{x+s} u(s) ds. \quad (50)$$

The exact solution is e^{-x} [11].

The effects of the multiplicity parameter r , the refinement level J , and thresholding with different thresholds are reported in Table 1 and Figure 1. The results confirm the theoretical claims and demonstrate the effectiveness of the method. Note that the L^2 error decreases as parameters r and J increase due to the rate of convergence $O(2^{-Jr})$. We compare the error of the proposed method, the Lagrange collocation method [10], Taylor collocation method [11], and Taylor polynomial method [26] in Table 2. Due to Table 2, our method is flexible than other methods, and without changing the multiplicity parameter r , we can improve the results. Figure 2 illustrate the effect of thresholding with different threshold parameters ε on the coefficient matrix. It can be seen that the number of matrix elements decreases when the threshold parameter increases.

Example 2. Let us consider the following VFIE:

$$\begin{aligned} u(x) &= \frac{e^{2x} - 1}{2} \cos(x) + e^x + \frac{1 - e^2}{2} \sin(x) \\ &+ \int_0^x e^s \cos(x) u(s) ds - \int_0^1 e^s \sin(x) u(s) ds. \end{aligned} \quad (51)$$

In Table 3 and Figure 3, we show the effect of the parameters r , J , and ε on sparsity and L_2 -error. It is obvious that increasing the parameters r and J reduces the error. A comparison of the proposed method with other methods such as Taylor method [26] and the Lagrange collocation [10] is reported in Table 4. In Figure 4, we illustrate the effect of thresholding on the coefficient matrix by taking different threshold ε when $r = 8$ and $J = 2$.

Example 3. Let us consider nonlinear VFIE (1) with

$$\begin{aligned} u(x) &= \sin(x) + (1 - e^{\sin(1)})x^3 - \frac{x^2 - \sin^2(x)}{4} \\ &- \int_0^x (x-s)u^2(s) ds - \int_0^1 x^3 \cos(s) e^{u(s)} ds. \end{aligned} \quad (52)$$

The exact solution of this equation is $u(x) = \sin(x)$.

In Figure 5, we illustrate the effect of parameter r and J on the L^2 -error and the error of approximation is plotted in Figure 6 taking $r = 7$ and $J = 2$.

Example 4. Consider the nonlinear VFIE as

$$u(x) = f(x) + \int_0^x \cos(x+s)u^2(s) ds + \int_0^1 \sin(s-x)u^2(s) ds, \quad (53)$$

with

$$\begin{aligned} f(x) &= \frac{(-2 \cos(2x) - \sin(2x))e^{2x}}{5} + \frac{2e^2 \sin(-1+x)}{5} \\ &+ \frac{e^2 \cos(-1+x)}{5} + \frac{\cos(x)}{5} + e^x - \frac{\sin(x)}{5}. \end{aligned} \quad (54)$$

(1) Choose a vector w_1 , such that $\|w_1\|_2 = 1$
 (2) For $j = 1, 2, \dots, m$ do
 (3) Compute $h_{i,j} = (\Lambda w_j, w_i)$ for $i = 1, 2, \dots, j$
 (4) $w_j = (\Lambda w_j, w_i)$ for $i = 1, 2, \dots, j$
 (5) $h_{j+1,j} = \|w_j\|_2$
 (6) If $h_{2j+1,j} = 0$ then stop
 (7) $w_{j+1} = (w_j/h_{j+1,j})$
 (8) End do

ALGORITHM 1: Arnoldi's algorithm.

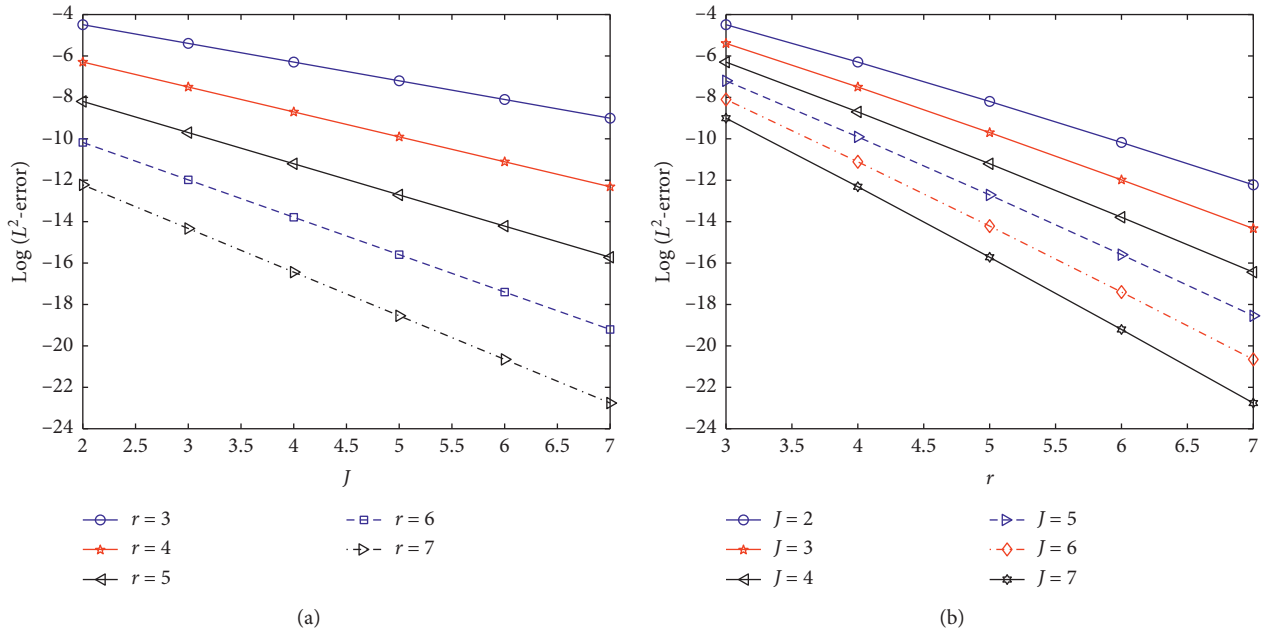


FIGURE 1: Effects of the refinement level J and multiplicity parameter r for Example 1.

(1) Compute $r_0 = D - \Lambda U_0$, $\beta = \|r_0\|_2$ and $w_1 = r_0/\beta$
 (2) Generate the Arnoldi basis and the matrix \tilde{H}_m using the Arnoldi algorithm starting with w_1
 (3) Compute y_m the minimizer of $\|\beta e_1 - \tilde{H}_m y\|_2$ and $U_m = U_0 + W_m y_m$
 (4) If satisfied then stop, else set $U_0 = U_m$ and go to 1

ALGORITHM 2: Restarted (GMRES).

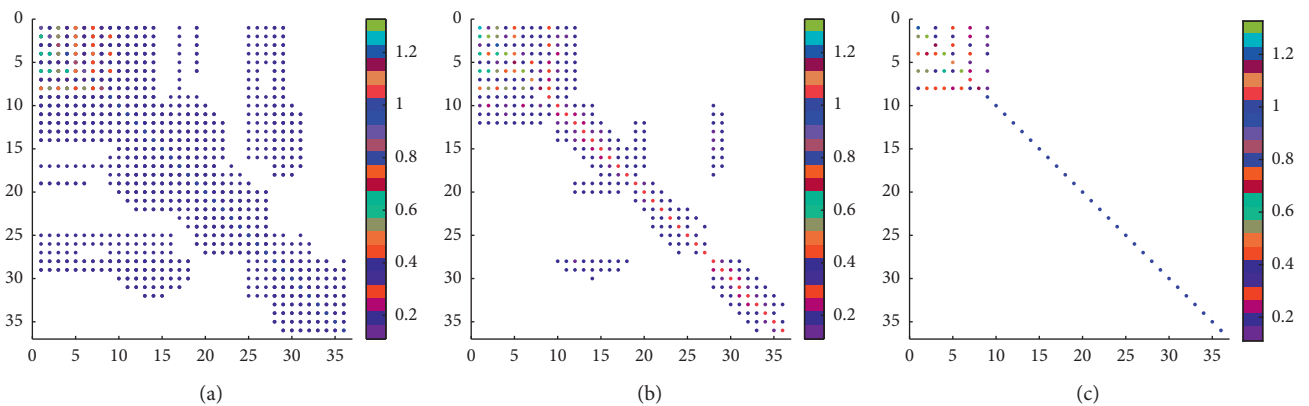


FIGURE 2: The effect of thresholding with threshold $\epsilon = 10^{-5}$ (left), $\epsilon = 10^{-3}$ (middle), and $\epsilon = 10^{-1}$ (right) on the coefficient matrix for Example 1.

TABLE 1: Effects of parameters r , J , and ε on sparsity and L_2 -error for Example 1.

r	J	Without thresholding		$\varepsilon = 10^{-5}$		$\varepsilon = 10^{-3}$		$\varepsilon = 10^{-1}$	
		S_ε	L_2 -error	S_ε	L_2 -error	S_ε	L_2 -error	S_ε	L_2 -error
5	2	0	$6.39e-9$	24.8	$1.20e-6$	54	$2.74e-4$	91	$2.96e-2$
	3	0	$2.00e-10$	59.8	$4.87e-6$	78.8	$2.78e-4$	96.5	$2.96e-2$
7	2	0	$5.97e-13$	43	$1.72e-6$	65.9	$1.38e-4$	97.5	$7.98e-3$
	3	0	$4.71e-15$	72.4	$2.65e-6$	85.4	$1.38e-4$	93.8	$7.98e-3$

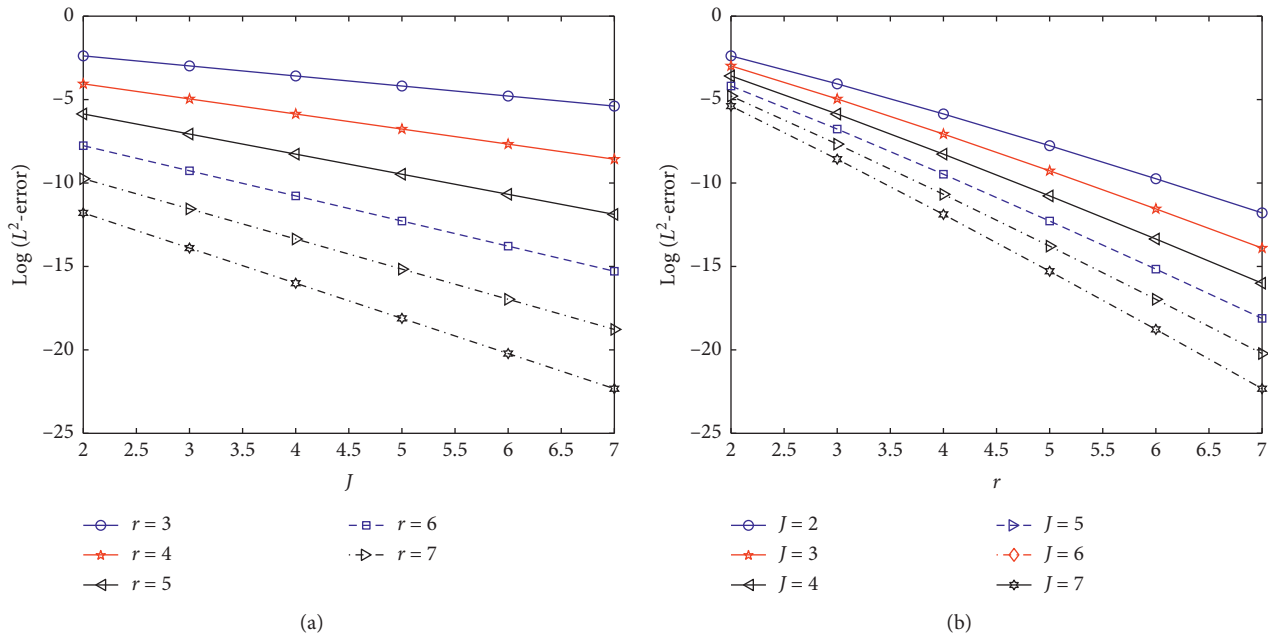


FIGURE 3: Effects of the refinement level J and multiplicity parameter r for Example 2.

TABLE 2: Comparison of the error for Example 1.

r	Presented method	Taylor collocation	Taylor polynomial method	Lagrange collocation
2	$1.55e-3$	$7.87e-2$	$3.41e-2$	$7.87e-2$
5	$6.39e-9$	$6.23e-5$	$3.68e-4$	$6.23e-5$
8	$4.63e-15$	$1.89e-8$	$1.24e-5$	$1.77e-7$
9	$4.01e-16$	$2.35e-8$	$3.46e-7$	$7.21e-6$

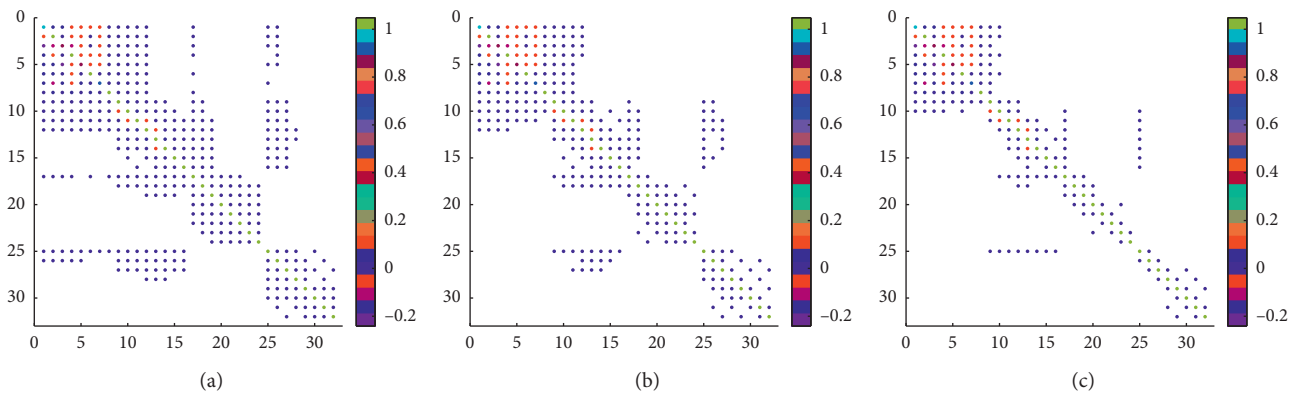


FIGURE 4: The effect of thresholding with threshold $\varepsilon = 10^{-5}$ (left), $\varepsilon = 10^{-4}$ (middle), and $\varepsilon = 10^{-3}$ (right) on the coefficient matrix for Example 2.

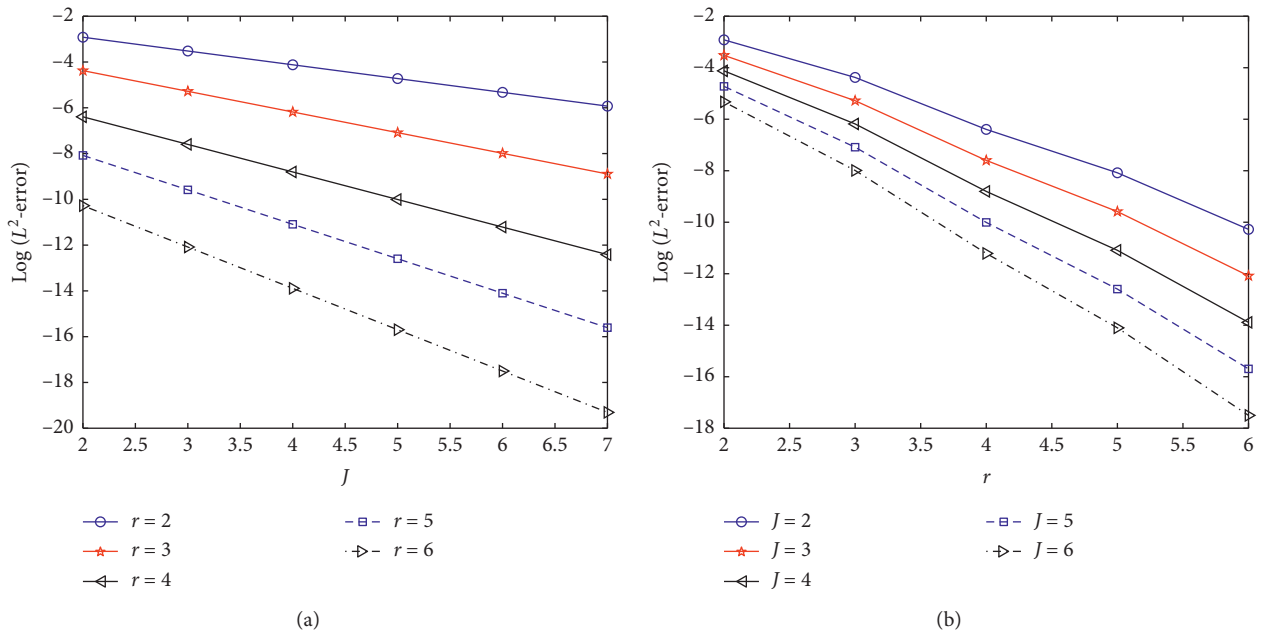


FIGURE 5: Effects of the refinement level J and multiplicity parameter r for Example 3.

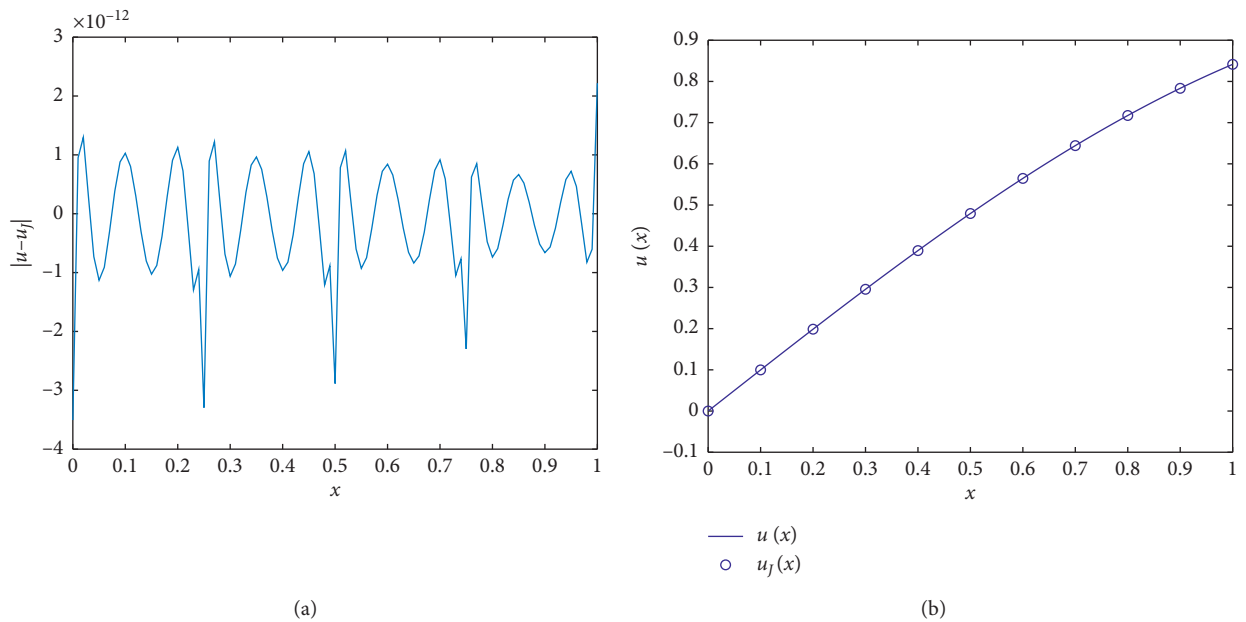


FIGURE 6: Error of Example 3.

TABLE 3: Effects of parameters r , J , and ε on sparsity and L_2 -error for Example 2.

r	J	Without thresholding		$\varepsilon = 10^{-6}$		$\varepsilon = 10^{-4}$		$\varepsilon = 10^{-2}$	
		S_ε	$L_2\text{-error}$	S_ε	$L_2\text{-error}$	S_ε	$L_2\text{-error}$	S_ε	$L_2\text{-error}$
5	2	0	$1.73e-8$	22.2	$1.00e-6$	51.5	$1.74e-4$	79.2	$5.15e-3$
	3	0	$5.43e-10$	58.1	$1.18e-6$	76.2	$1.74e-4$	91.9	$5.15e-3$
6	2	0	$1.81e-10$	32.5	$5.24e-10$	59.5	$7.81e-5$	82.1	$1.37e-2$
	3	0	$2.84e-12$	65.3	$1.02e-6$	81.7	$7.80e-5$	93.7	$1.37e-2$
7	2	0	$1.62e-12$	39.4	$6.78e-12$	65.1	$9.95e-5$	84.8	$1.84e-2$
	3	0	$1.27e-14$	71.3	$2.74e-7$	84.7	$9.95e-5$	95	$1.84e-2$

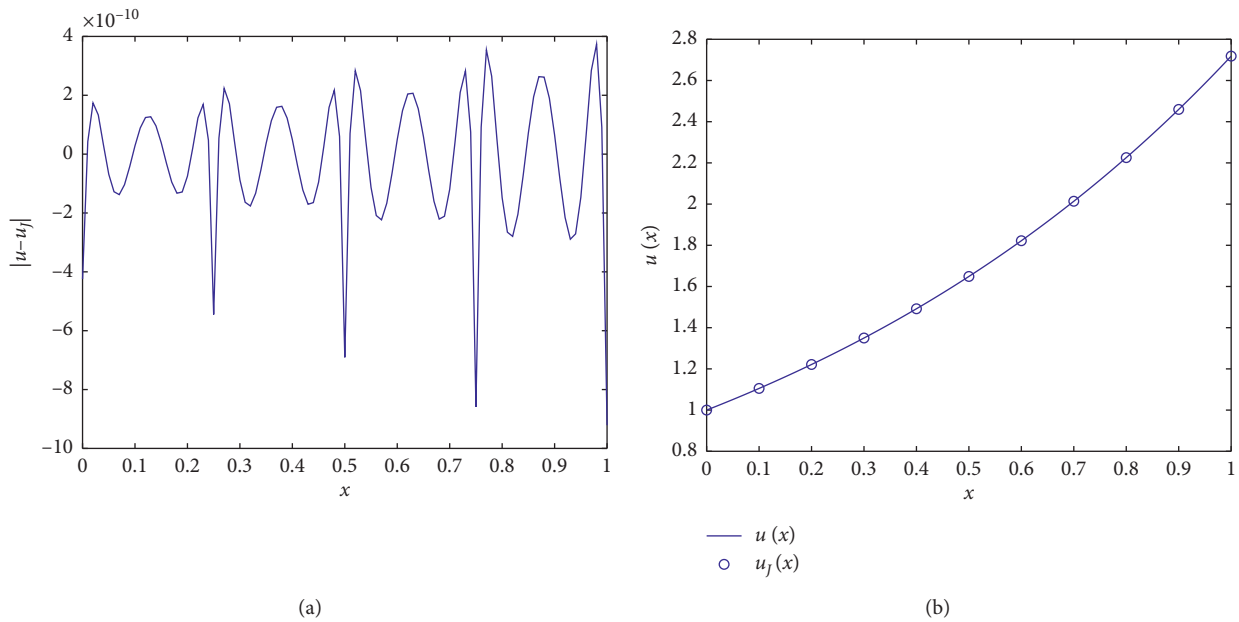


FIGURE 7: Error of Example 4.

TABLE 4: Comparison of the error for Example 2.

r	Presented method	Lagrange collocation	Taylor method
2	$4.17e-3$	$1.06e-2$	$4.51e-2$
5	$1.73e-8$	$1.12e-6$	$2.98e-4$
8	$1.27e-14$	$9.35e-7$	$6.14e-7$

TABLE 5: Effect of the refinement level J and multiplicity parameter r on L_2 -error for Example 4.

r	$J = 2$	$J = 3$	$J = 4$	$J = 5$	$J = 6$	$J = 7$
2	$4.18e-3$	$1.04e-3$	$2.61e-4$	$6.53e-5$	$1.68e-5$	$4.15e-6$
3	$8.77e-5$	$1.10e-5$	$1.67e-6$	$1.71e-7$	$2.19e-8$	$2.68e-9$
4	$1.38e-6$	$8.92e-8$	$5.09e-9$	$3.31e-10$	$2.11e-11$	$1.32e-12$
5	$1.73e-8$	$5.71e-10$	$1.79e-11$	$5.28e-13$	$1.61e-14$	$5.16e-16$
6	$1.81e-10$	$2.23e-12$	$4.64e-14$	$6.95e-16$	$1.15e-17$	$1.69e-19$

The exact solution is $u(x) = e^x$.

Table 5 is reported to show the efficiency and accuracy of the proposed method. We observe when the refinement level J and multiplicity parameter r increase, the L^2 -errors decrease. Figure 7 shows the error of proposed method on taking $r = 6$ and $J = 2$.

5. Conclusion

We have employed the multiwavelet Galerkin method to solve the Volterra–Fredholm integral equations. To this end, the Volterra and Fredholm operators are represented in multiwavelet bases. Applying this method leads to a linear or nonlinear system of algebraic equations. In the linear type, we obtain a new sparse system using thresholding due to the decay in the wavelet coefficients. The convergence analysis is investigated, and one can show that the rate of convergence is $O(2^{-Jr})$. The numerical examples illustrate the efficiency and accuracy of the method.

Data Availability

The raw/processed data required to reproduce these findings cannot be shared at this time due to legal or ethical reasons.

Conflicts of Interest

The author declares that there are no conflicts of interest.

Acknowledgments

This project was supported by researchers supporting project number RSP-2020/210, King Saud University, Riyadh, Saudi Arabia.

References

[1] B. G. Pachpatte, “On mixed volterra–fredholm type integral equations,” *Indian Journal of Pure and Applied Mathematics*, vol. 17, pp. 488–496, 1986.

- [2] L. Hacia, "On approximate solution for integral equations of mixed type," *ZAMM-Journal of Applied Mathematics and Mechanics*, vol. 76, pp. 415-416, 1996.
- [3] H. Brunner, "On the numerical solution of nonlinear Volterra-Fredholm integral equations by collocation methods," *SIAM Journal on Numerical Analysis*, vol. 27, no. 4, pp. 987-1000, 1990.
- [4] E. Banifatemi, M. Razzaghi, and S. Yousefi, "Two-dimensional Legendre wavelets method for the mixed volterra-fredholm integral equations," *Journal of Vibration and Control*, vol. 13, no. 11, pp. 1667-1675, 2007.
- [5] A.-M. Wazwaz, "A reliable treatment for mixed Volterra-Fredholm integral equations," *Applied Mathematics and Computation*, vol. 127, no. 2-3, pp. 405-414, 2002.
- [6] M. Ghasemi, M. Fardi, and R. Khoshsiar Ghaziani, "Solution of system of the mixed Volterra-Fredholm integral equations by an analytical method," *Mathematical and Computer Modelling*, vol. 58, no. 7-8, pp. 1522-1530, 2013.
- [7] A. Yildirim, "Homotopy perturbation method for the mixed Volterra-Fredholm integral equations," *Chaos, Solitons & Fractals*, vol. 42, no. 5, pp. 2760-2764, 2009.
- [8] S. A. Yousefi, A. Lotfi, and M. Dehghan, "He's variational iteration method for solving nonlinear mixed Volterra-Fredholm integral equations," *Computers & Mathematics with Applications*, vol. 58, no. 11-12, pp. 2172-2176, 2009.
- [9] B. N. Saray, "Sparse multiscale representation of Galerkin method for solving linear-mixed Volterra-Fredholm integral equations," *Mathematical Methods in the Applied Sciences*, vol. 43, no. 5, pp. 2601-2614, 2020.
- [10] K. Wang and Q. Wang, "Lagrange collocation method for solving Volterra-Fredholm integral equations," *Applied Mathematics and Computation*, vol. 219, no. 21, pp. 10434-10440, 2013.
- [11] K. Wang and Q. Wang, "Taylor collocation method and convergence analysis for the Volterra-Fredholm integral equations," *Journal of Computational and Applied Mathematics*, vol. 260, pp. 294-300, 2014.
- [12] A. Karamete and M. Sezer, "A Taylor collocation method for the solution of linear integro-differential equations," *International Journal of Computer Mathematics*, vol. 79, no. 9, pp. 987-1000, 2002.
- [13] M. Gülsu and M. Sezer, "Taylor collocation method for solution of systems of high-order linear Fredholm-Volterra integro-differential equations," *International Journal of Computer Mathematics*, vol. 83, no. 4, pp. 429-448, 2006.
- [14] F. Mirzaee, "Numerical solution of nonlinear fredholm-volterra integral equations via Bell polynomials," *Computational Methods for Differential Equations*, vol. 5, no. 2, pp. 88-102, 2017.
- [15] B. Alpert, G. Beylkin, R. Coifman, and V. Rokhlin, "Wavelet-like bases for the fast solution of second-kind integral equations," *SIAM Journal on Scientific Computing*, vol. 14, no. 1, pp. 159-184, 1993.
- [16] B. Alpert, G. Beylkin, D. Gines, and L. Vozovoi, "Adaptive solution of partial differential equations in multiwavelet bases," *Journal of Computational Physics*, vol. 182, no. 1, pp. 149-190, 2002.
- [17] N. Hovhannisyanyan and S. M. R. Schäfer, "Adaptive multi-resolution discontinuous Galerkin schemes for conservation laws," *Mathematics of Computation*, vol. 83, pp. 113-151, 2014.
- [18] B. N. Saray, "An efficient algorithm for solving volterra integro-differential equations based on Alpert's multi-wavelets Galerkin method," *Journal of Computational and Applied Mathematics*, vol. 348, pp. 453-465, 2019.
- [19] B. N. Saray, M. Lakestani, and M. Razzaghi, "Sparse representation of system of Fredholm integro-differential equations by using alpert multiwavelets," *Computational Mathematics and Mathematical Physics*, vol. 55, no. 9, pp. 1468-1483, 2015.
- [20] Y. Meyer, *Wavelets and Operators*, Cambridge University Press, Cambridge, England, 1992.
- [21] B. N. Saray and J. Manafian, "Sparse representation of delay differential equation of pantograph type using multiwavelets Galerkin method," *Engineering Computations*, vol. 35, no. 2, pp. 887-903, 2018.
- [22] K. E. Atkinson, *The Numerical Solution of Integral Equations of the Second Kind*, Cambridge University Press, Cambridge England, 1997.
- [23] Y. Saad and M. H. Schultz, "GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 3, pp. 856-869, 1986.
- [24] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, New Delhi, Delhi, 2003.
- [25] J. C. Goswami, A. K. Chan, and C. K. Chui, "On solving first-kind integral equations using wavelets on a bounded interval," *IEEE Transactions on Antennas and Propagation*, vol. 43, no. 6, pp. 614-622, 1995.
- [26] K. Maleknejad and Y. Mahmoudi, "Taylor polynomial solution of high-order nonlinear Volterra-Fredholm integro-differential equations," *Applied Mathematics and Computation*, vol. 145, no. 2-3, pp. 641-653, 2003.