Research Article

Numerical Techniques for Solving Linear Volterra Fractional Integral Equation

Safa’ Hamdan, Naji Qatanani, and Adnan Daraghmeh

Department of Mathematics, Faculty of Science, An-Najah National University, Nablus, State of Palestine

Correspondence should be addressed to Naji Qatanani; nqatanani@najah.edu and Adnan Daraghmeh; adn.daraghmeh@najah.edu

Received 2 May 2019; Accepted 24 July 2019; Published 20 December 2019

Academic Editor: Mustafa Inc

Copyright © 2019 Safa’ Hamdan et al. 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.

Two numerical techniques, namely, Haar Wavelet and the product integration methods, have been employed to give an approximate solution of the fractional Volterra integral equation of the second kind. To test the applicability and efficiency of the numerical method, two illustrative examples with known exact solution are presented. Numerical results show clearly that the accuracy of these methods are in a good agreement with the exact solution. A comparison between these methods shows that the product integration method provides more accurate results than its counterpart.

1. Introduction

Fractional integral equations of the second kind have attracted the attention of many scientists and researchers in recent years. These equations appear frequently in heat transformations and heat radiation, population growth model, biological species living together, porous media, rheology, control, and electro-chemistry (see [1–5] for more details). The concept of fractional calculus is now considered as a partial technique in many branches of science including physics (Oldham and Spanier [6]). Recently Srivastava et al. [7] gave the model of under actuated mechanical system with fractional order derivative and Sharma [8] studied advanced generalized fractional kinetic equation in Astrophysics. Caputo reformulated the more “classic” definition of the Riemann–Liouville fractional derivatives in order to use integer order initial conditions to solve his fractional order differential Equation [9]. Kowankar and Gangal reformulated the Riemann–Liouville fractional derivative in order to differentiate nowhere differentiable fractal functions [10]. Abdou et al. [11] have implemented the Toeplitz matrix and product Nystrom methods for solving singular integral equation with logarithmic kernel and Hilbert kernel. They concluded that these methods give a very accurate approximation for these types of kernels. In [12] Hamaydi and Qatanani have solved Linear Fuzzy Volterra Integral Equation. In addition, Hamdan [13] has employed several numerical methods for solving Volterra fractional integral equations. In recent years, numerous methods have been proposed for solving fractional Volterra integral Equations [2, 14, 15]. Lepik [16] has solved fractional integral equations by the Haar wavelet method. Fractional linear multistep methods have been employed by Lubich [17] to solve Abel’s Volterra integral equation of the second kind. Bruner [18] has solved Volterra integral and related functional equations by using the collocation method. Moreover, collocation method based on the orthogonal polynomials is presented to solve fractional integral Equations [19]. Other numerical methods on fractional integral equations are hybrid collocation [20], smoothing technique [15], piecewise constant orthogonal functions approximation [14], the Galerkin method [2], Berstein’s approximation [19, 21], the Simpson 3/8 rule method [22], mechanical quadrature [23], Legender Pseudo spectral [24], and iterative numerical method [25].

The fractional Volterra integral equation of the second kind under investigation is defined as follows:

\[
    u(x) = f(x) + \frac{1}{\Gamma(\alpha)} \int_{a}^{x} (x-t)^{\alpha-1} k(x,t)u(t)dt, \quad 0 < \alpha < 1, \tag{1}
\]

where \(k(x,t)\) is called the kernel or the nucleus of the integral equation. The function \(u(x)\) to be determined appears under the integral sign. The kernel \(k(x,t)\) and the function \(f(x)\) are given.
In this article, two numerical techniques, namely the Harr wavelet and the product integration method are implemented to solve the Volterra fractional integral Equation (1). A comparison between these methods is carried out by solving some test examples.

The paper is organized as follows: In Section 2 the Haar wavelet method is introduced. The product integration method used to approximate solution of the fractional integral equation is addressed in Section 3. The proposed methods are implemented using numerical examples with known analytical solution by applying MAPLE software in Section 4. Conclusions are given in Section 5.

2. The Haar Wavelet Method

The Haar functions are an orthogonal family of switched rectangular waveforms where amplitudes can differ from one function to another. They are defined in the interval [0, 1]. However, we are interested in defining these functions for the more general interval [A, B]. First, we define the integer \( M = 2^l \) (\( l \) is the maximal level of resolution). We divide the interval \([A, B]\) into \(2M\) subintervals of equal length; each subinterval has the length \( \Delta z = (B - A)/2M \). Next, two parameters are introduced: the dilation parameter \( i \) for which \( i = 0, 1, \ldots, l \) and the translation parameter \( c = 0, 1, \ldots, m - 1 \) (here the notation \( m = 2^l \) is introduced). The wavelet number \( j \) is identified as \( j = m + c + 1 \), \((m > c \text{ and } m \neq c)\).

The \( j^{th} \) Haar wavelet is defined as:

\[
h_j(x) = \begin{cases} 
1, & \tau_i \leq x \leq \tau_{i+1}, \\
-1, & \tau_{i+1} \leq x \leq \tau_{i+2}, \\
0, & \text{otherwise},
\end{cases}
\]  

where

\[
\tau_i = A + 2c\mu\Delta z, \\
\tau_{i+1} = A + (2c + 1)\mu\Delta z, \\
\tau_{i+2} = A + 2(c+1)\mu\Delta z, \\
\mu = \frac{M}{m},
\]

For \( j = 1 \) and \( x \in [0, 1] \), the function \( h_1(x) \) is the scaling function or the father wavelet for the family of the Haar wavelets which is defined as

\[
h_1(x) = \begin{cases} 
1, & 0 \leq x < 1, \\
0, & \text{otherwise}.
\end{cases}
\]  

For \( j = 2 \) and \( x \in [0, 1] \), the function \( h_2(x) \) is the mother wavelet for the family of the Haar wavelet which is defined as

\[
h_2(x) = \begin{cases} 
1, & 0 \leq x < \frac{1}{2}, \\
-1, & \frac{1}{2} \leq x < 1, \\
0, & \text{otherwise}.
\end{cases}
\]  

For \( j = 3 \) and \( x \in [0, 1] \), the function \( h_3(x) \) is defined as

\[
h_3(x) = \begin{cases} 
1, & 0 \leq x < \frac{1}{4}, \\
-1, & \frac{1}{4} \leq x < \frac{1}{2}, \\
0, & \text{otherwise}.
\end{cases}
\]  

For \( j = 4 \) and \( x \in [0, 1] \), the function \( h_4(x) \) is defined as

\[
h_4(x) = \begin{cases} 
1, & 0 \leq x < \frac{1}{8}, \\
-1, & \frac{1}{8} \leq x < \frac{1}{4}, \\
0, & \text{otherwise}.
\end{cases}
\]

Assume that

\[
t = \frac{1}{4} + \frac{2}{3}, \quad 4 \leq x < \frac{1}{2},
\]

\[
\frac{1}{2} \leq x < \frac{1}{4}.
\]

In fact Table 1 shows the index computations for Harr basis functions. The Haar wavelet series expansion of a given function \( g(x) \) is given as [24]

\[
g(x) = \sum_{j=1}^{2M} b_j h_j(x).
\]  

where \( x \in [A, B] \) and \( b_j \) are the wavelet coefficients. There are many methods for computing \( b_j \), one of these is the collocation method.

The collocation points are defined as

\[
x_k = A + (k - 0.5)\Delta z, \quad k = 1, 2, \ldots, 2M.
\]

Substituting Equation (9) into Equation (8) gives the discrete version, i.e.,

\[
g(x_k) = \sum_{j=1}^{2M} b_j h_j(x_k).
\]  

It is convenient to put Equation (10) into the matrix form as

\[
g = Hb
\]

where \( g \) and \( b \) are \( 2M \)-dimensional row vectors and the Haar coefficients \( H(j, k) = h_j(x_k) \) is the element of a \( 2M \times 2M \) matrix.

Now, in virtue of Equation (9) and upon substituting Equation (10) into Equation (1) we get

\[
g(x_k) = f(x_k) + \frac{1}{\Gamma(\alpha)} \int_0^{x_k} (x_k - t)^{\alpha-1} k(x_k, t) g(t) dt
\]

or

\[
\sum_{j=1}^{2M} b_j h_j(x_k) = f(x_k) + \frac{1}{\Gamma(\alpha)} \int_0^{x_k} (x_k - t)^{\alpha-1} k(x_k, t) \sum_{j=1}^{2M} b_j h_j(t) dt
\]

\[
f(x_k) = \sum_{j=1}^{2M} b_j h_j(x_k) - \frac{1}{\Gamma(\alpha)} \int_0^{x_k} (x_k - t)^{\alpha-1} k(x_k, t) \sum_{j=1}^{2M} b_j h_j(t) dt
\]

\[
f(x_k) = \sum_{j=1}^{2M} \left[ h_j(x_k) - \frac{1}{\Gamma(\alpha)} \int_0^{x_k} (x_k - t)^{\alpha-1} k(x_k, t) h_j(t) dt \right]
\]

Table 1: Index computations for Haar basis function.

<table>
<thead>
<tr>
<th>( j )</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>2</th>
<th>2</th>
<th>2</th>
<th>2</th>
<th>2</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>( c )</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>( j = 2^3 + c + 1 )</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
</tr>
</tbody>
</table>
\[ u_j(x_k) = \frac{1}{\Gamma(a)} \int_0^{x_k} (x_k - t)^{a-1} k(x_k, t) h_j(t) \, dt \] (16)

then Equation (15) becomes

\[ f(x_k) = \sum_{j=1}^{2M} b_j [h_j(x_k) - u_j(x_k)], \quad k = 1, 2, \ldots, 2M. \] (17)

The matrix form of Equation (17) is

\[ F = b[H - U], \] (18)

where \( U(j, k) = u_j(x_k), F(k) = f(x_k). \)

The solution of Equation (18) is

\[ b = F((H - U))^{-1}. \] (19)

Upon using Equation (2), then Equation (16) can be rewritten in the following form

\[
\begin{align*}
U(j, k) &= 0, \quad x_k < \tau_j, \\
U(j, k) &= \frac{1}{\Gamma(a)} \int_{\tau_j}^{x_k} (x_k - t)^{a-1} k(x_k, t) \, dt, \quad \tau_j \leq x \leq \tau_{j+1}, \\
U(j, k) &= \frac{1}{\Gamma(a)} \int_{\tau_j}^{\tau_{j+1}} (x_k - t)^{a-1} k(x_k, t) \, dt - \int_{\tau_j}^{x_k} (x_k - t)^{a-1} k(x_k, t) \, dt, \quad \tau_j \leq x \leq \tau_{j+1}, \\
U(j, k) &= \frac{1}{\Gamma(a)} \int_{\tau_j}^{\tau_{j+1}} (x_k - t)^{a-1} k(x_k, t) \, dt - \int_{\tau_j}^{x_{j+1}} (x_k - t)^{a-1} k(x_k, t) \, dt, \quad \tau_j \leq x_k.
\end{align*}
\] (20)

**3. The Product Integration Method**

We introduce the product integration method that can be used to solve linear Volterra fractional integral equation of the second kind.

To use the product integration method as a numerical technique \([3, 10]\), we consider the linear Volterra integral equation of the second kind which has the general form:

\[ g(x) = f(x) + \lambda \int_a^x k(x, t) g(t) \, dt, \quad a \leq x \leq b, \] (21)

where \( k(x, t) \) is the kernel (and is known), \( \lambda \) is a constant parameter and \( g(x) \) is unknown function to be determined. This method is based on factoring of the singularity of the kernel \( k(x, t) \) as follows:

\[ k(x, t) = p(x, t) G(x, t), \] (22)

where \( G(x, t) \) is well-behaved function and \( p(x, t) \) is badly behaved function.

Set Equation (22) into Equation (21) yields

\[ g(x) = f(x) + \lambda \int_a^x p(x, t) G(x, t) g(t) \, dt, \quad a \leq x \leq b. \] (23)

First, we decompose the interval \([a, b]\) into \( N \) subintervals \( h_i \) where:

\[ h_i = x_{i+1} - x_i, \quad i = 0, 1, 2, \ldots, N - 1 \] (24)

and

\[ a = x_0 < x_1 < \cdots < x_N = b. \] (25)

If we use the \( N \)-points quadrature rule and collocate Equation (23) at the nodes \( \{x_i\}_{i=1}^N \cup \{0\} \), we get

**Table 2:** The exact and numerical solutions using the Haar wavelet method with \( N = 64 \).

| \( x_k \) | Exact solution \( g(x) = x^{\frac{1}{2}} - 1 \) | Numerical solution \( g_n(x) \) | Error : \(|g - g_n|\) |
|---|---|---|---|
| 0.0078 | -0.9116116523 | -0.9116094999 | 1.0702e-05 |
| 0.1172 | -0.6576734015 | -0.6576324164 | 4.0985e-04 |
| 0.2266 | -0.5240141808 | -0.5239489931 | 6.5188e-05 |
| 0.3359 | -0.4203988440 | -0.4203067536 | 9.2090e-05 |
| 0.4453 | -0.3326826092 | -0.3325562333 | 2.1843e-04 |
| 0.5547 | -0.2552265445 | -0.2550525941 | 3.5635e-04 |
| 0.6641 | -0.1850996993 | -0.1848551078 | 2.4459e-04 |
| 0.7734 | -0.1205470450 | -0.1201906958 | 3.5635e-04 |
| 0.8828 | -0.0604189763 | -0.0598738640 | 5.4511e-04 |
| 0.9922 | -0.0039139093 | -0.0030264584 | 8.8745e-04 |

\[ g(x) = f(x) + \lambda \sum_{j=0}^N w_j G(x_j, t_j) g(t_j), \quad i = 0, 1, 2, \ldots, N. \] (26)

Using Lagrange interpolation polynomial, Equation (26) can be written as:

\[ g(x) = f(x) + \lambda \sum_{j=0}^N w_j G(x_j, t_j) g(t_j), \quad i = 0, 1, 2, \ldots, N. \] (27)

It follows that

\[ \int_a^x p(x, t) G(x, t) g(t) \, dt = \sum_{j=0}^N \int_{t_j}^{t_{j+1}} p(x, t) G(x, t) g(t) \, dt \]

\[ = \sum_{j=0}^N \int_{t_j}^{t_{j+1}} p(x, t) \left[ \frac{G(x, t_j) g(t_j)}{h_j} + \frac{G(x, t_{j+1}) g(t_{j+1})}{h_j} \right] \, dt \]

\[ = \sum_{j=0}^N w_j G(x, t_j) g(t_j), \] (29)

where

\[ w_0 = \int_{t_j}^{t_{j+1}} p(x, t) \left( \frac{t_{j+1} - t}{h_0} \right) \, dt, \quad \text{for } j = 0 \] (30)

\[ w_{j+1} = \int_{t_j}^{t_{j+1}} p(x, t) \left( \frac{t_{j+1} - t}{h_j} \right) \, dt + \int_{t_{j+1}}^{t_{j+2}} p(x, t) \left( \frac{t - t_{j+1}}{h_{j+1}} \right) \, dt, \quad \text{for } j = 1, \ldots, i - 1, \] (31)
The solution of Volterra fractional integral equation by Haar wavelet method

\[
F = \{f(x_0), f(x_1), \ldots, f(x_N)\}^T, \quad (39)
\]
\[
\tilde{G}_n = [\tilde{g}(x_0), \tilde{g}(x_1), \ldots, \tilde{g}(x_N)] = \tilde{g}(b)\tilde{b}. \quad (40)
\]

**Algorithm 1: Numerical realization using the Haar wavelet method.**

1: Input:
   - \(A, B\): \([A, B]\) is the interval for the solution function.
   - \(I\): The maximal level of resolution.
   - \(i\): The dilation parameter.
   - \(c\): The translation parameter.
   - \(f(x)\): The function of the integral equation.
   - \(k(x, t)\): The kernel function.
   - \(h_1(x)\): The scaling function.
   - \(\lambda\): A constant parameter.

2: Calculate:
   - The quantity \(M = 2^I\).
   - \(N = 2M\)
   - The length of each subintervals \(\Delta x = \frac{B-A}{N}\).
   - \(m = 2^i\)
   - The wavelet number \(j = m + c + 1\).

3: Calculate: The collocation points \(x_k = A + (k - 0.5)\Delta x\).
4: Calculate: \(h_1(x_k), u_j(x_k)\) and \(f(x_k)\).
5: Calculate:
   - The matrix \([H - U]\).
   - Calculate \(b = F([H - U]^{-1})\).

6: Determine the solution of the linear system \(g = bH\).
1: Input:
- \(a, b: [a, b]\) is the interval for the solution function.
- \(N\): The number of subdivisions of \([a, b]\).
- \(f(x)\): The function of the integral equation.
- \(G(x, t), p(x, t)\): The kernel functions.
- \(\lambda\): Is a constant parameter.
- Set \(x_0 = a\) and \(x_N = b\).

2: Calculate:
- \(h = \frac{b - a}{N}\)
- \(t_j = x_j = a + ih\)
- \(\hat{h}_j = x(i + 1) - x_i\)

3: Calculate: \(w_{i0}, w_{ij}, w_{ii}\)

\[
\begin{align*}
& w_{i0} = \int_{t_i}^{t_j} p(x_i, t)\left(\frac{t_i - t}{h_i}\right) dt, \quad \text{for } j = 0 \\
& w_{ij} = \int_{t_i}^{t_{j+1}} p(x_i, t)\left(\frac{t_{j+1} - t}{h_i}\right) dt + \int_{t_j}^{t_i} p(x_i, t)\left(\frac{t_j - t}{h_j}\right) dt, \quad \text{for } j = 1, \cdots, i - 1 \\
& w_{ij} = \int_{t_{j+1}}^{t_i} p(x_i, t)\left(\frac{t_i - t}{h_j}\right) dt, \quad \text{for } j = i
\end{align*}
\]

4: Solve the recurrence relation:

\[
\hat{g}(x_i) = f(x_i) + \lambda \sum_{j=0}^{N} w_{ij} G(x_i, t_j) \hat{g}(t_j), \quad i = 1, 2, \cdots, N
\]

or we can also solve the linear system of algebraic equations

\[
\hat{G}_n = B^{-1} F
\]

Algorithm 2: Numerical realization using the product integration method.

Here \(B\) is a lower triangular matrix \((N + 1) \times (N + 1)\) and \((1 - \lambda w_j G(x_i, x_j)) \neq 0\).

3.1. Uniqueness of a Solution. To prove the uniqueness of the solution, we can apply the Cauchy–Minkowski inequality (see [11] for more details) and get \(|\lambda| \leq 1\) which represents a sufficient condition to have a unique solution, and the value of \(|g_i|\) satisfies the inequality:

\[
|g_i| \leq \frac{|f_i|}{1 - \lambda}. \tag{41}
\]

4. Numerical Examples and Results

In this section, in order to examine the accuracy of the proposed methods, we solve two numerical examples of fractional Volterra integral equations. Moreover, the numerical results will be compared with exact solution.

Example 1. Consider the linear Volterra fractional integral equation of the second kind:

\[
g(x) = x^{1/2} - \frac{3}{4} \sqrt{\pi} x^2 + \frac{8}{3} \sqrt{\pi} x^{3/2} - 1 + \frac{1}{\sqrt{\pi}} \int_0^x (x - t)^{-1/2} 2t g(t) dt. \tag{42}
\]

The analytical solution of Equation (42) is:

\[
g(x) = x^{1/2} - 1. \tag{43}
\]

We implement Algorithm 1 to solve Equation (42) using the Haar wavelet method. Table 2 displays the exact and numerical results using the Haar wavelet method for Equation (42) with \(\Delta x = 0.0156\) and \(N = 64\). The maximum error with \(N = 64\) is 8.8745 e−04. Figure 1(a) compares both the exact and numerical solutions for the Volterra fractional integral Equation (42). Moreover, Figure 1(b) shows the absolute error between exact and numerical solutions. Now, we implement Algorithm 2 to solve Equation (42) using the Product Integration Method. We obtain the following results:

4.1. W Function. If \(p(x_i, t) = (x_i - t)^{-1/2}\), then we have

\[
w_{i0} = \frac{1}{h_i} \int_{t_i}^{t_j} p(x_i, t)(t_i - t) dt, \quad \text{for } j = 0, \tag{44}
\]

\[
w_{ij} = \frac{1}{h_i} \int_{t_i}^{t_j} (x_i - t)^{-1/2}(h_i - t) dt, \quad \text{for } j = 0, \tag{45}
\]

\[
w_{ij} = \frac{1}{h_i} \int_{t_i}^{t_j} (ih_i - t)^{-1/2}(h_i - t) dt, \quad \text{for } j = 1, \cdots, i - 1, \tag{46}
\]

\[
w_{ij} = \frac{2}{3} h_{ij} \left[3 i^{1/2} + 2 \left(i - 1\right)^{3/2} - i^{3/2}\right], \tag{47}
\]

\[
w_{ij} = \int_{t_i}^{t_{j+1}} p(x_i, t)\left(\frac{t_{j+1} - t}{h_j}\right) dt \tag{48}
\]

\[
+ \int_{t_{j+1}}^{t_i} p(x_i, t)\left(\frac{t - t_{j+1}}{h_{ij}}\right) dt, \quad j = 1, \cdots, i - 1,
\]

\[
w_{ij} = \int_{t_i}^{t_{j+1}} (x_i - t)^{-1/2}\left(\frac{(j + 1)h_i - t}{h_i}\right) dt \tag{49}
\]

\[
+ \int_{t_i}^{t_{j+1}} (x_i - t)^{-1/2}\left(\frac{(j + 1)h_i - t}{h_i}\right) dt,
\]

\[
w_{ij} = \frac{4}{3} h_{ij} \left[(i - j)^{3/2} + \left[(i + j)^{3/2} - 2(i - j)^{3/2}\right]\right], \tag{50}
\]

\[
w_{ii} = \frac{1}{h_i} \int_{t_{i-1}}^{t_i} p(x_i, t)(t - t_{i-1}) dt, \quad j = i, \tag{51}
\]

\[
w_{ii} = \frac{1}{h_i} \int_{(i-1)}^{t_i} (x_i - t)^{-1/2}(t - (i - 1)h) dt, \tag{52}
\]

\[
w_{ii} = \frac{1}{h_i} \int_{(i-1)}^{t_i} (ih_i - t)^{-1/2}(t - (i - 1)h) dt, \tag{53}
\]

\[
w_{ii} = \frac{4}{3} h_i^{1/2}. \tag{54}
\]
From above equations we obtain the linear system of algebraic equations:

$$G_n = B^{-1} F.$$ (55)

Table 3 displays the exact and the numerical results using the product integration method for Equation (42) with $h = 0.0156$. The maximum error with $N = 64$ is $4.7435e^{-04}$. Figure 2(a) compares both the analytical and numerical solutions for Equation (42) with $N = 64$. Moreover, Figure 2(b) shows the absolute error between exact and numerical solutions.

**Example 2.** Consider the Abel’s integral equation of the second kind:

$$g(x) = \frac{1}{3} x^3 - x + 0.2399x^{7/3} + 0.8399x^{4/3} + 0.3733 \int_0^x (x-t)^{-2/3}g(t)dt.$$ (56)

The exact solution of Equation (56) is:

$$g(x) = \frac{1}{3} x^3 - x.$$ (57)

Applying Algorithms 1 to Equation (56), we obtain the following results. Table 4 displays the exact and the numerical results using the Haar wavelet method for Equation (56) with $\Delta z = 0.0156$ and $N = 64$. The maximum error with $N = 64$ is $1.1113e^{-03}$. Figure 3(a) compares both the exact and numerical solutions for the Abel’s integral equation (56). Moreover, Figure 3(b) shows the absolute error between exact and numerical solutions. Applying Algorithm 2 to Equation (56) using the product integration method, we get

$$w_{ij} = \int_{t_i}^{t_j} p(x_i, t) \left( \frac{(t_{j+1} - t)}{h_j} \right) dt + \int_{t_{j+1}}^{t_i} p(x_j, t) \left( \frac{(t - t_{j+1})}{h_{j+1}} \right) dt, \quad j = 1, \ldots, i - 1, (62)$$

$$w_{ij} = \frac{3}{4} h^{1/3} [4t^{1/3} + 3(i - 1)^{4/3} - t^{4/3} + 3], \quad i = 1, \ldots, N$$ (63)

**Figure 2:** A comparison and absolute error between exact and numerical solutions in Example 1 with $N = 64$. (a) A comparison. (b) Absolute error.
\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|}
\hline
\(x_n\) & Exact solution \(g(x) = x^{1/2} - 1\) & Numerical solution \(g_n(x)\) & Error \(\|g - g_n\|\) \\
\hline
0.0078 & -0.0077921549 & -0.0083481389 & 5.5598e^{-04} \\
0.1172 & -0.1126098632 & -0.1131662257 & 5.5636e^{-04} \\
0.2266 & -0.2094523111 & -0.2100916889 & 6.3938e^{-04} \\
0.3359 & -0.2983194986 & -0.2990367942 & 7.1730e^{-04} \\
0.4453 & -0.3792114257 & -0.380030241 & 7.9160e^{-04} \\
0.5547 & -0.4521280924 & -0.4529904969 & 8.6240e^{-04} \\
0.6641 & -0.5170694986 & -0.517999650 & 9.3015e^{-04} \\
0.7734 & -0.5740356445 & -0.5750303556 & 9.9471e^{-04} \\
0.8828 & -0.6230265299 & -0.6240817729 & 1.0552e^{-03} \\
0.9922 & -0.6640421549 & -0.6651534675 & 1.1113e^{-03} \\
\hline
\end{tabular}
\caption{The exact and numerical solutions using the Haar wavelet method with \(N = 64\).}
\end{table}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3.pdf}
\caption{A comparison and absolute error between exact and numerical solutions in Example 2 with \(N = 64\). (a) A comparison. (b) Absolute error.}
\end{figure}

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|}
\hline
\(x_n\) & Exact solution \(g(x) = x^{1/2} - 1\) & Numerical solution \(g_n(x)\) & Error \(\|g - g_n\|\) \\
\hline
0 & 0 & 0 & 0 \\
0.125 & -0.1197916666 & -0.1197818969 & 9.7697e^{-05} \\
0.250 & -0.2291666666 & -0.2291501145 & 1.6552e^{-05} \\
0.375 & -0.3281250000 & -0.3280928056 & 2.6794e^{-05} \\
0.500 & -0.4166666666 & -0.4166237294 & 4.2937e^{-05} \\
0.625 & -0.4947916666 & -0.4947244207 & 6.7246e^{-04} \\
0.750 & -0.5625000000 & -0.5623979179 & 1.0208e^{-04} \\
0.875 & -0.6197916666 & -0.6196416593 & 1.5001e^{-04} \\
1 & -0.6666666666 & -0.6664528164 & 2.1385e^{-04} \\
\hline
\end{tabular}
\caption{The exact and numerical solutions using the product integration method with \(N = 64\).}
\end{table}

\begin{align}
\sum_{j=0}^{n-1} \int_{h \cdot (j+1)}^{h \cdot j} (x - t)^{-2/3} \left( \frac{t - h \cdot j}{h} \right) dt = & \sum_{j=0}^{n-1} \int_{h \cdot (j+1)}^{h \cdot j} (x - t)^{-2/3} \left( \frac{t - h \cdot j}{h} \right) dt, \quad j = i, \quad (64) \\
\sum_{j=0}^{n-1} \int_{h \cdot (j+1)}^{h \cdot j} (x - t)^{-2/3} \left( \frac{t - h \cdot j}{h} \right) dt = & \sum_{j=0}^{n-1} \int_{h \cdot (j+1)}^{h \cdot j} (x - t)^{-2/3} \left( \frac{t - h \cdot j}{h} \right) dt, \quad (65) \\
\sum_{j=0}^{n-1} \int_{h \cdot (j+1)}^{h \cdot j} (x - t)^{-2/3} \left( \frac{t - h \cdot j}{h} \right) dt = & \sum_{j=0}^{n-1} \int_{h \cdot (j+1)}^{h \cdot j} (x - t)^{-2/3} \left( \frac{t - h \cdot j}{h} \right) dt, \quad j = i, \quad (66) \\
\sum_{j=0}^{n-1} \int_{h \cdot (j+1)}^{h \cdot j} (x - t)^{-2/3} \left( \frac{t - h \cdot j}{h} \right) dt = & \sum_{j=0}^{n-1} \int_{h \cdot (j+1)}^{h \cdot j} (x - t)^{-2/3} \left( \frac{t - h \cdot j}{h} \right) dt, \quad (67)
\end{align}
Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References


