A Higher-Order Numerical Scheme for Two-Dimensional Nonlinear Fractional Volterra Integral Equations with Uniform Accuracy

Zi-Qiang Wang, Qin Liu and Jun-Ying Cao *

School of Data Science and Information Engineering, Guizhou Minzu University, Guiyang 550025, China; ziqiang100@126.com (Z.-Q.W.); liuqin0628@126.com (Q.L.)
* Correspondence: caojunying1000@126.com

Abstract: In this paper, based on the modified block-by-block method, we propose a higher-order numerical scheme for two-dimensional nonlinear fractional Volterra integral equations with uniform accuracy. This approach involves discretizing the domain into a large number of subdomains and using biquadratic Lagrangian interpolation on each subdomain. The convergence of the high-order numerical scheme is rigorously established. We prove that the numerical solution converges to the exact solution with the optimal convergence order $O(h_1^{2-a} + h_2^{4-\beta})$ for $0 < a, \beta < 1$. Finally, experiments with four numerical examples are shown, to support the theoretical findings and to illustrate the efficiency of our proposed method.

Keywords: nonlinear fractional Volterra integral equations; high-order numerical scheme; convergence analysis; optimal convergence order

1. Introduction

Volterra integral equations (VIEs) have significant applications in various fields of applied science and engineering, such as superfluidity, elasticity, electromagnetism, electrostatics, potential theory, geophysics, etc. Singular and weakly singular integral equations are of particular interest, since they are used to solve inverse boundary value problems whose domains are fractal curves, where classical calculus cannot be used. Abel equations and other fractional-order integral equations have been studied extensively and are used in modeling various phenomena in biophysics, viscoelasticity and electrical circuits [1,2]. In this paper, we consider using the biquadrature formula to solve the following two-dimensional nonlinear VIEs with a fractional-order weakly singular kernel:

$$u(x, y) = f(x, y) + \int_a^x \int_c^y \kappa(x, y, s, t, u(s, t)) \frac{(x-s)^a(y-t)^\beta}{(x-s)^a(y-t)^\beta} ds \, dt, \quad (x, y) \in D, 0 < a, \beta < 1,$$

(1)

where $f(x, y)$ and $\kappa(x, y, s, t, u(s, t))$ are given continuous functions defined on $D = [a, b] \times [c, d], \Omega = D \times D \times R$, and $u(x, y)$ is an unknown function defined on $D$. In addition, we assume that the VIEs (1) have a smooth solution $u(x, y)$ and that $\kappa(x, y, s, t, u(s, t))$ satisfies the following condition

$$|\kappa(x, y, s, t, u_1(s, t)) - \kappa(x, y, s, t, u_2(s, t))| \leq L|u_1(s, t) - u_2(s, t)|, \quad L > 0.$$

(2)

The classical block-by-block method is an efficient numerical algorithm for VIEs. A general block-by-block method for one-dimensional linear VIEs with a nonsingular kernel function was given in [3]. In [4], systems of one-dimensional nonlinear VIEs with a nonsingular kernel function were solved by the block-by-block method. A new high-order numerical scheme for fractional ordinary differential equations was given in [5], called the modified block-by-block method. In [6], radial basis functions were used to solve for the
second kind of nonlinear VIEs in which the kernel function satisfies the Lipschitz condition. The authors of [7] used the Runge–Kutta method and the block-by-block method to solve the second kind of nonlinear VIEs with a continuous kernel. In [8], a new transformation was used to solve the VIEs by using a Laplace transform. The Bernstein approximation method was used for VIEs of the third kind in [9]. In [10], the spectral collocation method with graded meshes was used to solve the second kind of VIEs with a weakly singular kernel. Two-dimensional VIEs were studied by an Euler-type numerical scheme in [11]. A method based on two-dimensional Euler polynomials combined with the Gauss–Jacobi quadrature formula was used to solve two-dimensional VIEs with fractional-order weakly singular kernels in [12]. General linear methods were implemented with a variable step size for VIEs with a sufficiently smooth kernel function, and the corresponding MATLAB code was developed in [13]. VIEs with a nonlinear weakly singular kernel function were solved by an hp-version collocation method in conjunction with Jacobi polynomials in [14]. The optimal homotopy asymptotic method was used for linear and nonlinear two-dimensional VIEs in [15]. In [16], an iterated multi-Galerkin method was used to solve VIEs with a weakly singular kernel, and the proof of convergence rates was given by the projected methods. Multivariate Bernstein polynomials were used to solve multidimensional linear and nonlinear VIEs with fractional weakly singular kernel functions in [17]. In [18], a Jacobi spectral collocation method was proposed for a class of nonlinear VIEs with kernels of the form \( x^\beta (z - x)^{-\alpha} g(y(x)) \), where \( \alpha \in (0, 1), \beta > 0 \) and \( g(y) \) is a nonlinear function. Numerical solutions for weakly singular VIEs were presented based on Chebyshev and Legendre pseudo-spectral Galerkin methods in [19]. Some more recent developments are shown in [20–22]. The study of numerical solutions for high-dimensional VIEs with singular nonlinear kernel functions remains an important research issue. To date, there have been few reports on high-order numerical algorithms for two-dimensional VIEs with singular nonlinear kernel functions.

In this paper, we construct a new high-order numerical solution for two-dimensional nonlinear fractional VIEs by using the techniques of [5] with uniform accuracy. To avoid degeneracy near the two boundary layers, the proposed scheme couples the solutions at the two boundary layers. Thus, no smaller meshes are needed to achieve the sharp numerical order. Such coupling is not required in the other subdomains. The convergence analysis is based on a novel technology that couples the ideas of [5] and the Gronwall inequality. Hence, the optimal convergence order \( O(h_\alpha^{k-\alpha} + h_\beta^{k-\beta}) \) for \( 0 < \alpha, \beta < 1 \) can be achieved for a sufficiently smooth solution and a general nonlinear kernel function.

This paper is arranged as follows. In Section 2, the higher-order numerical scheme is proposed. The estimation of the truncation errors of the constructed higher-order numerical scheme is given in Section 3. A convergence analysis of the higher-order numerical scheme is given in Section 4. In Section 5, four numerical experiments are presented to illustrate the efficiency of the the high-order numerical approach and to support our theoretical findings. Finally, some conclusions arising from this work are drawn in Section 6.

2. Higher-Order Numerical Scheme of Two-Dimensional Nonlinear Fractional VIEs

In this section, we consider the approximate evaluation of two-dimensional nonlinear fractional VIEs. In order to construct the numerical scheme of Equation (1), the region \( D \) is divided into \( 2M \times 2N \) equal subdomains with size \( h_x = \frac{b-a}{2M}, h_y = \frac{d-c}{2N} \), where \( M \) and \( N \) are positive integers. We denote \( x_i = a + ih_x y_j = c + jh_y, i = 0, 1, \ldots, 2M; j = 0, 1, \ldots, 2N \), and the numerical solution of formula (1) at point \( (x_i, y_j) \) is denoted by \( u_{i,j} \), where \( u_{i,j} = f(x_i, 0), u_{0,j} = f(0, y_j) \). For convenience of narration, we let \( \kappa_{i,j}(s, t, u(s,t)) = \kappa(x_i, y_j, s, t, u(s,t)) \), \( f_{i,j} = f(x_i, y_j) \).

Firstly, we propose a high-order scheme for the nonlinear fractional VIEs. Let \( \varphi_{i,j}(s), k = 0, 1, 2; i \in \mathbb{N} \) and \( \phi_{k,j}(t), k = 0, 1, 2; j \in \mathbb{N} \) be the basis functions of quadratic interpolation polynomials at points \( x_i, x_{i+1}, x_{i+2} \) and \( y_j, y_{j+1}, y_{j+2} \), respectively, where \( \varphi_{k,j}(s) \) and \( \phi_{k,j}(t), k = 0, 1, 2, \) are defined as follows.
\[
\varphi_{0,i}(s) = \frac{(s-x_{i+1})(s-x_{i+2})}{2h_y^2}, \quad \varphi_{1,i}(s) = \frac{(s-x_i)(s-x_{i+2})}{-h_x^2}, \quad \varphi_{2,i}(s) = \frac{(s-x_i)(s-x_{i+1})}{2h_y^2};
\]
\[
\varphi_{0,j}(t) = \frac{(t-y_{j+1})(t-y_{j+2})}{2h_x^2}, \quad \varphi_{1,j}(t) = \frac{(t-y_j)(t-y_{j+2})}{-h_y^2}, \quad \varphi_{2,j}(t) = \frac{(t-y_j)(t-y_{j+1})}{2h_x^2}.
\]

We estimate that \(u(x, y)\) at point \((x_1, y_1)\) has the form

\[
u(x_1, y_1) = f_{1,1} + \int_a^x \int_c^{y_1} \frac{\chi_{1,1}(s, t, u(s, t))}{(x_1-s)^a(y_1-t)^b} dtds
\]
\[
\approx f_{1,1} + \int_a^x \int_c^{y_1} \frac{1}{(x_1-s)^a(y_1-t)^b} \sum_{i=0}^{2} \sum_{j=0}^{2} \varphi_{i,0}(s) \varphi_{j,0}(t) \chi_{1,1}(x_i, y_j, u_{i,j}) dtds
\]
\[
= f_{1,1} + \sum_{i=0}^{2} \sum_{j=0}^{2} \omega_{i,0}^{(1)} \omega_{j,0}^{(1)} \chi_{1,1}(x_i, y_j, u_{i,j}),
\]
with
\[
\omega_{1,0}^{(1)} = \int_a^{x_1} \frac{\varphi_{0,0}(s)}{(x_1-s)^a} ds, i = 0, 1, 2; \quad \omega_{1,0}^{(1)} = \int_c^{y_1} \frac{\varphi_{j,0}(t)}{(y_1-t)^b} dt, j = 0, 1, 2.
\]

To compute \(u(x_2, y_1), u(x_1, y_2)\) and \(u(x_2, y_2)\), we use the following approximation:

\[
u(x_2, y_1) = f_{2,1} + \int_a^x \int_c^{y_1} \frac{\chi_{2,1}(s, t, u(s, t))}{(x_2-s)^a(y_1-t)^b} dtds \approx f_{2,1} + \sum_{i=0}^{2} \sum_{j=0}^{2} A_{2}^{(1)} \omega_{i,0}^{(1)} \chi_{2,1}(x_i, y_j, u_{i,j}),
\]
\[
u(x_1, y_2) = f_{1,2} + \int_a^x \int_c^{y_2} \frac{\chi_{1,2}(s, t, u(s, t))}{(x_1-s)^a(y_2-t)^b} dtds \approx f_{1,2} + \sum_{i=0}^{2} \sum_{j=0}^{2} \omega_{i,0}^{(1)} A_{2}^{(1)} \chi_{1,2}(x_i, y_j, u_{i,j}),
\]
\[
u(x_2, y_2) = f_{2,2} + \int_a^x \int_c^{y_2} \frac{\chi_{2,2}(s, t, u(s, t))}{(x_2-s)^a(y_2-t)^b} dtds \approx f_{2,2} + \sum_{i=0}^{2} \sum_{j=0}^{2} A_{2}^{(1)} \omega_{j,0}^{(1)} \chi_{2,2}(x_i, y_j, u_{i,j}),
\]
where
\[
A_{2}^{(1)} = \int_a^{x_2} \frac{\varphi_{0,0}(s)}{(x_2-s)^a} ds, i = 0, 1, 2; \quad \tilde{A}_{2}^{(1)} = \int_c^{y_2} \frac{\varphi_{j,0}(t)}{(y_2-t)^b} dt, j = 0, 1, 2.
\]

Note that computing \(u_{1,1}\) through (3) requires the values of \(\chi\) (or indirectly, the values of \(u\)) at \(x_1, x_2\) and \(y_1, y_2\). Particularly, the dependence of \(u_{1,1}\) on \(\chi_{2,1}, \chi_{1,2}\) and \(\chi_{2,2}\) means that (3), (5)–(7) must be solved simultaneously with the scheme.

We further estimate \(u(x_{2m+1}, y_i), m = 1, \cdots, M - 1\) and \(u(x_i, y_{2n+1}+1), l, r = 1, 2, n = 1, \cdots, N - 1\), assuming \(u_{i,r}, i = 0, 1, \cdots, 2m\) and \(u_{l,r}, j = 0, 1, \cdots, 2n\) are already known. For \(u(x_{2m+1}, y_1)\), we have

\[
u(x_{2m+1}, y_1) = f_{2m+1,1} + \int_a^x \int_c^{y_1} \frac{\chi_{2m+1,1}(s, t, u(s, t))}{(x_{2m+1} - s)^a(y_1-t)^b} dtds
\]
\[
+ \sum_{i=1}^{m} \sum_{k=0}^{2} \sum_{q=0}^{2} \int_a^{x_{2i+1}} \int_c^{y_1} \frac{\chi_{2m+1,1}(s, t, u(s, t))}{(x_{2m+1} - s)^a(y_1-t)^b} dtds
\]
\[
= f_{2m+1,1} + \sum_{i=1}^{m} \sum_{j=1}^{2} \sum_{k=0}^{2} \sum_{q=0}^{2} \int_a^{x_{2i+1}} \int_c^{y_1} \frac{\chi_{2m+1,1}(s, t, u(s, t))}{(x_{2m+1} - s)^a(y_1-t)^b} dtds
\]
\[
+ \sum_{i=1}^{m} \sum_{j=1}^{2} \sum_{k=0}^{2} \sum_{q=0}^{2} \int_a^{x_{2i+1}} \int_c^{y_1} \frac{\chi_{2m+1,1}(s, t, u(s, t))}{(x_{2m+1} - s)^a(y_1-t)^b} dtds.
\]

Fractal Fract. 2022, 6, 314
3 of 27
where \( \hat{\omega}^{0,0}_{1} \) is in (4) and
\[
\omega^{k,0}_{2m+1} = \int_{a}^{x_{2m+1}} (x_{2m+1} - s)^{-\alpha} \varphi_{k,0}(s) ds, k = 0, 1, 2, \tag{10}
\]
\[
A^{k,i}_{2m+1} = \int_{x_{2i-1}}^{x_{2i+1}} (x_{2m+1} - s)^{-\alpha} \varphi_{k,2i-1}(s) ds, k = 0, 1, 2; i = 1, \cdots, m. \tag{11}
\]

We estimate \( u(x_{2m+2}, y_{1}) \) and \( u(x_{2m+1}, y_{2}) \), \( l = 1, 2 \) using the following approximations
\[
u(x_{2m+2}, y_{1}) = f_{2m+1} + \sum_{i=1}^{m} \int_{x_{2i-1}}^{x_{2i+2}} \int_{c}^{y_{1}} \frac{\kappa_{2m+1,2}(s, t, u(s, t))}{(x_{2m+1} - s)^{\alpha}(y_{2} - t)^{\beta}} dt ds \tag{12}
\]
\[
u(x_{2m+1}, y_{2}) = f_{2m+2} + \sum_{i=1}^{m} \int_{x_{2i-1}}^{x_{2i+2}} \int_{c}^{y_{2}} \frac{\kappa_{2m+2,2}(s, t, u(s, t))}{(x_{2m+1} - s)^{\alpha}(y_{2} - t)^{\beta}} dt ds \tag{13}
\]
where \( \hat{\omega}^{0,0}_{1} \) is in (4), \( \hat{\omega}^{0,0}_{2m+1} \) and \( A^{k,i}_{2m+1} \) are in (10) and (11), respectively, and
\[
A^{k,i}_{2m+2} = \int_{x_{2i}}^{x_{2i+2}} (x_{2m+2} - s)^{-\alpha} \varphi_{k,2i}(s) ds, k = 0, 1, 2; i = 0, 1, \cdots, m. \tag{15}
\]

Next we use the same method to estimate \( u(x_{r}, y_{2n+1}) \), \( l, r = 1, 2 \) and directly obtain
\[
u(x_{1}, y_{2n+1}) = f_{1,2n+1} + \int_{a}^{x_{1}} \int_{c}^{y_{1}} \frac{\kappa_{1,2n+1}(s, t, u(s, t))}{(x_{1} - s)^{\alpha}(y_{2n+1} - t)^{\beta}} dt ds \tag{16}
\]
\[
\begin{align*}
\frac{d^2}{dt^2} \varphi(t) &= \sum_{k=1}^{N} A_{k,0} \frac{d}{dt} \varphi(t) + \sum_{k=1}^{N} A_{k,1} \varphi(t), \\
\frac{d^2}{dt^2} \psi(t) &= \sum_{k=1}^{N} A_{k,0} \frac{d}{dt} \psi(t) + \sum_{k=1}^{N} A_{k,1} \psi(t),
\end{align*}
\]
\[
\sum = \int_a^{x_1} \int_c^{y_i} (x_{2m+1} - s)^{-a} \phi_{k,0}(s)\phi_{q,0}(t)K_{2m+1,2n+1}(x_k, y_q, u_{k,q}) dtds
\]

\[
= \sum_{k=0}^{a} \sum_{q=0}^{b} \omega_{2m+1,k}^{0} \hat{\omega}_{2n+1,q}^{0} K_{2m+1,2n+1}(x_k, y_q, u_{k,q}), \quad (24)
\]

where \(\omega_{2m+1}^{0}\) is shown in (10) and \(\omega_{2n+1}^{0}\) is defined in (20).

For \(I_2\), we can obtain

\[
I_2 = \sum_{k=0}^{a} \sum_{q=0}^{b} \omega_{2m+1,k}^{0} \hat{A}_{2n+1,q}^{0} K_{2m+1,2n+1}(x_k, y_q, u_{k,q}) \quad (25)
\]

where \(\omega_{2m+1}^{0}\) is shown in (10) and \(A_{2n+1}^{0}\) is defined in (21).

For \(I_3\), we have

\[
I_3 = \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=0}^{a} \sum_{q=0}^{b} \omega_{2m+1,k}^{0} \hat{A}_{2n+1,q}^{0} K_{2m+1,2n+1}(x_k, y_q, u_{k,q}) \quad (26)
\]

where \(A_{2m+1}^{0}\) and \(\hat{A}_{2n+1}^{0}\) are shown in (11) and (20), respectively.

For \(I_4\), along the same lines, we have

\[
I_4 = \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=0}^{a} \sum_{q=0}^{b} \omega_{2m+1,k}^{0} \hat{A}_{2n+1,q}^{0} K_{2m+1,2n+1}(x_k, y_q, u_{k,q}) \quad (27)
\]

where \(A_{2m+1}^{0}\) is shown in (11) and \(A_{2n+1}^{0}\) is defined in (21).

Bringing (24)-(27) into (23), we obtain

\[
\begin{align*}
&u_{2m+1,2n+1} = f_{2m+1,2n+1} + \sum_{k=0}^{a} \sum_{q=0}^{b} \omega_{2m+1,k}^{0} \hat{\omega}_{2n+1,q}^{0} K_{2m+1,2n+1}(x_k, y_q, u_{k,q}) \\
&\quad + \sum_{j=1}^{n} \sum_{k=0}^{a} \sum_{q=0}^{b} \omega_{2m+1,k}^{0} \hat{A}_{2n+1,q}^{0} K_{2m+1,2n+1}(x_k, y_q, u_{k,q}) \\
&\quad + \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=0}^{a} \sum_{q=0}^{b} A_{2m+1,i}^{0} \hat{A}_{2n+1,j}^{0} K_{2m+1,2n+1}(x_k, y_q, u_{k,q}) \\
&\quad + \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=0}^{a} \sum_{q=0}^{b} A_{2m+1,i}^{0} \hat{A}_{2n+1,j}^{0} K_{2m+1,2n+1}(x_k, y_q, u_{k,q})
\end{align*}
\]
\begin{align*}
+ \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=0}^{2} \sum_{l=0}^{2} A_{2m+1,i}^{q,j} A_{2n+1}^{q,j} \kappa_{2m+1,2n+1} \left( x_{2i-1+k, y_{2j-1+q, u_{2j-1+k,2j-1+q}}} \right). 
\end{align*}
\tag{28}

Next, we construct the numerical scheme for \(u(x_{2m+2}, y_{2n+1})\). We first split the integration domain, and then use the piecewise biquadratic interpolation to approximate the calculation in the corresponding subdomains. Therefore, the numerical scheme of \(u(x_{2m+2}, y_{2n+1})\) can be given by the following

\begin{align*}
&u(x_{2m+2}, y_{2n+1}) = f_{2m+2,2n+1} + \sum_{i=0}^{m} \sum_{j=0}^{n} \sum_{k=0}^{2} \sum_{l=0}^{2} A_{2m+2,i}^{k,j} A_{2n+1}^{q,j} \kappa_{2m+2,2n+1} \left( x_{2i+k, y_{2j-1+q, u_{2i+k,2j-1+q}}} \right)
+ \frac{1}{2} \sum_{i=0}^{m} \sum_{j=0}^{n} \int_{x_{2i-1}}^{x_{2i+2}} \int_{y_{2j-1}}^{y_{2j+1}} \kappa_{2m+2,2n+1} \left( s, t, u(s, t) \right) \frac{dtds}{(x_{2m+2} - s)^a (y_{2n+1} - t)^a}
+ \frac{1}{2} \sum_{i=0}^{m} \sum_{j=0}^{n} \int_{x_{2i-1}}^{x_{2i+2}} \int_{y_{2j-1}}^{y_{2j+1}} \kappa_{2m+2,2n+1} \left( s, t, u(s, t) \right) \frac{dtds}{(x_{2m+2} - s)^a (y_{2n+1} - t)^a}
\approx f_{2m+2,2n+1} + \sum_{i=0}^{m} \sum_{j=0}^{n} \sum_{k=0}^{2} \sum_{l=0}^{2} A_{2m+2,i}^{k,j} A_{2n+1}^{q,j} \kappa_{2m+2,2n+1} \left( x_{2i+k, y_{2j-1+q, u_{2i+k,2j-1+q}}} \right),
\end{align*}
\tag{29}

where \(A_{2m+2,i}^{k,j} A_{2n+1}^{q,j} \) and \(A_{2m+1,i}^{k,j} A_{2n+2}^{q,j} \) are in (15), (20) and (21), respectively.

As a consequence, \(u(x_{2m+1}, y_{2n+2})\) and \(u(x_{2m+2}, y_{2n+2})\) can be approximated as follows

\begin{align*}
u(x_{2m+1}, y_{2n+2}) &= f_{2m+1,2n+2} + \sum_{j=0}^{n} \sum_{k=0}^{2} \sum_{l=0}^{2} \omega_{2m+1}^{k,j} A_{2n+2}^{k,j} \kappa_{2m+1,2n+2} \left( x_{2i+k, y_{2j+q, u_{2i+k,2j+q}}} \right)
+ \frac{1}{2} \sum_{j=0}^{n} \sum_{k=0}^{2} \sum_{l=0}^{2} \omega_{2m+1}^{k,j} A_{2n+2}^{k,j} \kappa_{2m+1,2n+2} \left( x_{2i+k, y_{2j+q, u_{2i+k,2j+q}}} \right)
+ \frac{1}{2} \sum_{j=0}^{n} \sum_{k=0}^{2} \sum_{l=0}^{2} \omega_{2m+1}^{k,j} A_{2n+2}^{k,j} \kappa_{2m+1,2n+2} \left( x_{2i+k, y_{2j+q, u_{2i+k,2j+q}}} \right),
\end{align*}
\tag{30}

where \(\omega_{2m+1}^{k,j} A_{2m+1,i}^{k,j} A_{2n+2}^{q,j} \) and \(A_{2m+2}^{q,j} \) are shown in (10), (11), (15) and (22), respectively.

To summarize, by combining (3), (5)–(7), (9), (12)–(14), (16)–(19) and (28)–(31) we compose the high-order numerical scheme of Equation (1) as follows

\(u_{1,1} = f_{1,1} + \sum_{i=0}^{2} \sum_{j=0}^{2} \omega_{1}^{l,j} A_{1}^{l,j} \kappa_{1,1} \left( x_{i, y_{j, u_{1,i}}} \right),\)

\(u_{2,1} = f_{2,1} + \sum_{i=0}^{2} \sum_{j=0}^{2} A_{2}^{l,j} \kappa_{1,2} \left( x_{i, y_{j, u_{2,i}}} \right),\)

\(u_{1,2} = f_{1,2} + \sum_{i=0}^{2} \sum_{j=0}^{2} \omega_{1}^{l,j} A_{1}^{l,j} \kappa_{1,2} \left( x_{i, y_{j, u_{1,i}}} \right),\)

\(u_{2,2} = f_{2,2} + \sum_{i=0}^{2} \sum_{j=0}^{2} A_{2}^{l,j} \kappa_{2,2} \left( x_{i, y_{j, u_{2,i}}} \right),\)
\[
\begin{align*}
U_{2m+1,1} &= f_{2m+1,1} + \sum_{k=0}^{2} \omega_{2m+1}^{k,0} \hat{Q}^{q,0}_{2m+1,1} (x_k, y_q, U_{k,q}) \\
&\quad + \sum_{i=1}^{m} \sum_{k=0}^{2} \sum_{q=0}^{2} A^{k,i}_{2m+1} \hat{Q}^{q,0}_{1} K_{2m+1,1} (x_{2i-1+k, y_q, U_{2i-1+k,q}}), \\
U_{2m+2,1} &= f_{2m+2,1} + \sum_{i=1}^{m} \sum_{k=0}^{2} \sum_{q=0}^{2} A^{k,i}_{2m+2} \hat{Q}^{q,0}_{1} K_{2m+2,1} (x_{2i+k, y_q, U_{2i+k,q}}), \\
U_{2m+1,2} &= f_{2m+1,2} + \sum_{k=0}^{2} \omega_{2m+1}^{k,0} \hat{Q}^{q,0}_{2m+1,2} (x_k, y_q, U_{k,q}) \\
&\quad + \sum_{i=1}^{m} \sum_{k=0}^{2} \sum_{q=0}^{2} A^{k,i}_{2m+1} \hat{Q}^{q,0}_{2} K_{2m+1,2} (x_{2i-1+k, y_q, U_{2i-1+k,q}}), \\
U_{2m+2,2} &= f_{2m+2,2} + \sum_{i=1}^{m} \sum_{k=0}^{2} \sum_{q=0}^{2} A^{k,i}_{2m+2} \hat{Q}^{q,0}_{2} K_{2m+2,2} (x_{2i+k, y_q, U_{2i+k,q}}), \\
U_{1,2n+1} &= f_{1,2n+1} + \sum_{k=0}^{2} \omega_{1}^{k,0} \hat{Q}^{q,0}_{2n+1} K_{1,2n+1} (x_k, y_q, U_{k,q}) \\
&\quad + \sum_{i=1}^{n} \sum_{k=0}^{2} \sum_{q=0}^{2} A^{k,i}_{2n+1} \hat{Q}^{q,0}_{1} K_{1,2n+1} (x_{2i-1+k, y_q, U_{2i-1+k,q}}), \\
U_{2,2n+1} &= f_{2,2n+1} + \sum_{k=0}^{2} \omega_{2}^{k,0} \hat{Q}^{q,0}_{2n+1} K_{2,2n+1} (x_k, y_q, U_{k,q}) \\
&\quad + \sum_{i=1}^{n} \sum_{k=0}^{2} \sum_{q=0}^{2} A^{k,i}_{2n+1} \hat{Q}^{q,0}_{2} K_{2,2n+1} (x_{2i-1+k, y_q, U_{2i-1+k,q}}), \\
U_{1,2n+2} &= f_{1,2n+2} + \sum_{i=0}^{n} \sum_{j=0}^{2} \sum_{q=0}^{2} \omega_{1}^{k,0} \hat{Q}^{q,0}_{j} K_{1,2n+2} (x_k, y_{2j+q}, U_{k,2j+q}) , \\
U_{2,2n+2} &= f_{2,2n+2} + \sum_{j=0}^{n} \sum_{j=0}^{2} \sum_{q=0}^{2} \omega_{2}^{k,0} \hat{Q}^{q,0}_{j} K_{2,2n+2} (x_k, y_{2j+q}, U_{k,2j+q}), \\
U_{2m+1,2n+1} &= f_{2m+1,2n+1} + \sum_{k=0}^{2} \omega_{2m+1}^{k,0} \hat{Q}^{q,0}_{2m+1,1} K_{2m+1,2n+1} (x_k, y_q, U_{k,q}) \\
&\quad + \sum_{i=1}^{m} \sum_{k=0}^{2} \sum_{q=0}^{2} A^{k,i}_{2m+1} \hat{Q}^{q,0}_{2} K_{2m+1,2n+1} (x_{2i-1+k, y_q, U_{2i-1+k,q}}) \\
&\quad + \sum_{i=1}^{m} \sum_{k=0}^{2} \sum_{q=0}^{2} \omega_{2m+1}^{k,0} \hat{Q}^{q,0}_{2} K_{2m+1,2n+1} (x_{2i-1+k, y_q, U_{2i-1+k,q}}) \\
&\quad + \sum_{i=1}^{m} \sum_{j=0}^{n} \sum_{k=0}^{2} \sum_{q=0}^{2} A^{k,i}_{2m+1} \hat{Q}^{q,0}_{j} K_{2m+1,2n+1} (x_{2i-1+k, y_{2j+q}, U_{2i-1+k,2j+1+q}}), \\
U_{2m+2,2n+1} &= f_{2m+2,2n+1} + \sum_{i=0}^{m} \sum_{k=0}^{2} \sum_{q=0}^{2} \omega_{2m+2}^{k,0} \hat{Q}^{q,0}_{2m+2,1} K_{2m+2,2n+1} (x_k, y_q, U_{k,q}) \\
&\quad + \sum_{i=0}^{m} \sum_{j=0}^{n} \sum_{k=0}^{2} \sum_{q=0}^{2} A^{k,i}_{2m+2} \hat{Q}^{q,0}_{j} K_{2m+2,2n+1} (x_{2i+k, y_{2j+q}, U_{2i+k,2j+1+q}}), \\
U_{2m+1,2n+2} &= f_{2m+1,2n+2} + \sum_{i=0}^{m} \sum_{j=0}^{n} \sum_{k=0}^{2} \sum_{q=0}^{2} \omega_{2m+1}^{k,0} \hat{Q}^{q,0}_{j} K_{2m+1,2n+2} (x_k, y_{2j+q}, U_{k,2j+q}) \\
&\quad + \sum_{i=0}^{m} \sum_{j=0}^{n} \sum_{k=0}^{2} \sum_{q=0}^{2} A^{k,i}_{2m+1} \hat{Q}^{q,0}_{j} K_{2m+1,2n+2} (x_{2i-1+k, y_{2j+q}, U_{2i-1+k,2j+q}})
\end{align*}
\]
\[ u_{2m+2,2n+2} = f_{2m+2,2n+2} + \sum_{i=0}^{m} \sum_{j=0}^{n} \sum_{k=0}^{2} A_{2m+2}^{kj} A_{2n+2}^{kj} K_{2m+2,2n+2} (x_{2i+k}, y_{2j+q}, u_{2i+k,2j+q}). \]

3. Estimation of the Truncation Errors

Now, we analyze the estimation of the truncation error of (32). We define the truncation error at the point \((x_i, y_j)\) by

\[ r_{i,j} := u(x_i, y_j) - \bar{u}_{i,j}, \]

where \(\bar{u}_{i,j}\) is an approximate value of \(u(x_i, y_j)\), which is substituted into the exact solution of (32). For example, we define \(\bar{u}_{i,j}\) at the point \((x_{2m+1}, y_{2n+1})\) as follows

\[ \bar{u}_{2m+1,2n+1} = f_{2m+1,2n+1} + \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=0}^{2} A_{2m+1}^{kj} A_{2n+1}^{kj} K_{2m+1,2n+1} (x_{k}, y_{j}, u(x_{k}, y_{j})). \]

For ease of notation, let \(\hat{\alpha}_j^2 \phi_{x} \) be the truncation error defined in (33). Then, the estimation of \(r_{i,j}\) is as follows.

Lemma 1. Let \(r_{i,j}\) be the truncation error defined in (33). If \(k(\cdot, \cdot, \cdot, u(\cdot, \cdot)) \in C^4([a, b] \times [c, d])\), then it holds that

\[ |r_{i,j}| \leq C(t^4_{x} + h^4_{y} - \beta), \]

where \(C\) depends only on \(\alpha, \beta, G_1, G_2\), and \(G_1, G_2\) are defined as follows:

\[ G_1 = \max_{x,s \in [a,b], y,t \in [c,d]} (|\partial_3^4 k(x, y, s, t, u(s, t))|, |\partial_4^4 k(x, y, s, t, u(s, t))|), \]

\[ G_2 = \max_{x,s \in [a,b], y,t \in [c,d]} (|\partial_4^4 k(x, y, s, t, u(s, t))|, |\partial_4^4 k(x, y, s, t, u(s, t))|). \]

Proof. Firstly, we estimate the truncation error \(r_{2m+1,2n+1}\). According to (33), we define the truncation error at the point \((x_{2m+1}, y_{2n+1})\). Combining (28), (33) and (34), we have

\[ \int_a^{x_{2m+1}} \int_y^{y_{2n+1}} \left( x_{2m+1} - s \right)^{-\alpha} \left( y_{2n+1} - t \right)^{-\beta} K_{2m+1,2n+1} (s, t, u(s, t)) \]

\[ - \sum_{k=0}^{2} \sum_{q=0}^{2} \phi_{k,0}(s) \phi_{q,0}(t) K_{2m+1,2n+1} (x_{k}, y_{q}, u(x_{k}, y_{q}))) dt ds \]

\[ + \sum_{j=1}^{n} \int_a^{y_{2j+1}} \int_{y_{2j-1}}^{y_{2j+1}} \left( x_{2m+1} - s \right)^{-\alpha} \left( y_{2n+1} - t \right)^{-\beta} K_{2m+1,2n+1} (s, t, u(s, t)) \]

\[ - \sum_{k=0}^{2} \sum_{q=0}^{2} \phi_{k,0}(s) \phi_{q,2j-1}(t) K_{2m+1,2n+1} (x_{k}, y_{2j-1+q}, u(x_{k}, y_{2j-1+q}))) dt ds \]
\[
+ \sum_{j=1}^{m} \int_{x_{2i-1}}^{x_{2i+1}} \int_{c}^{x_{2m+1}} (x_{2m+1} - s)^{-\alpha} (y_{2n+1} - t)^{-\beta} [K_{2m+1,2n+1}(s,t,u(s,t))] \, dt \, ds \\
- 2 \sum_{k=0}^{m} \int_{2i}^{2i+2} \phi_{k,2i-1}(s) \phi_{q,0}(t) K_{2m+1,2n+1}(x_{2i-1+k},y_q,u(x_{2i-1+k},y_q)) \, dt \, ds \\
+ \sum_{j=1}^{m} \sum_{j=1}^{n} \int_{x_{2i-1}}^{x_{2i+1}} \int_{2j-1}^{2j+1} (x_{2m+1} - s)^{-\alpha} (y_{2n+1} - t)^{-\beta} [K_{2m+1,2n+1}(s,t,u(s,t))] \\
- 2 \sum_{k=0}^{m} \int_{2i}^{2i+2} \phi_{k,2i-1}(s) \phi_{q,2j-1}(t) K_{2m+1,2n+1}(x_{2i-1+k},y_{2j-1+q},u(x_{2i-1+k},y_{2j-1+q})) \, dt \, ds \\
= \int_{a}^{x_{1}} \int_{c}^{x_{1}} (x_{2m+1} - s)^{-\alpha} (y_{2n+1} - t)^{-\beta} R_1 \, dt \, ds \\
+ \sum_{j=1}^{n} \int_{y_{2j-1}}^{y_{2j+1}} \frac{(x_{2m+1} - s)^{-\alpha}}{(y_{2n+1} - t)^{-\beta}} R_2 \, dt \\
+ \sum_{j=1}^{m} \sum_{j=1}^{n} \int_{2j-1}^{2j+1} (x_{2m+1} - s)^{-\alpha} (y_{2n+1} - t)^{-\beta} R_3 \, dt \\
+ \sum_{k=0}^{m} \sum_{q=0}^{n} \int_{2i}^{2i+2} \phi_{k,2i-1}(s) \phi_{q,2j-1}(t) K_{2m+1,2n+1}(x_{2i-1+k},y_{2j-1+q},u(x_{2i-1+k},y_{2j-1+q})) \, dt \, ds \\
\sum_{j=1}^{2m+1} + \sum_{j=1}^{2m+1} + \sum_{j=1}^{2m+1} + \sum_{j=1}^{2m+1} (37)
\]

Using Taylor’s theorem, one can obtain that for all \((s,t) \in [a, x_1] \times [c, y_1] \),

\[
R_1 = \frac{1}{3!} \frac{\partial^3 K_{2m+1,2n+1}(\xi_1(s), t, u(\xi_1(s), t))}{3!} \prod_{k=0}^{2} (s - x_k) \\
+ \frac{2}{3!} \sum_{k=0}^{2} \frac{\phi_{k,0}(s)}{3!} \frac{\partial^3 K_{2m+1,2n+1}(x_k, \eta_1(t), u(x_k, \eta_1(t)))}{3!} \prod_{q=0}^{2} (t - y_q),
\]

where \((\xi_1(s), \eta_1(t)) \in [a, x_1] \times [c, y_1] \). For all \((s,t) \in [a, x_1] \times [y_{2j-1}, y_{2j+1}] \), there exists \((\xi_2(s), \eta_2(t)) \in [a, x_1] \times [y_{2j-1}, y_{2j+1}] \), such that

\[
R_2 = \frac{1}{3!} \frac{\partial^3 K_{2m+1,2n+1}(\xi_2(s), t, u(\xi_2(s), t))}{3!} \prod_{k=0}^{2} (s - x_k) \\
+ \frac{2}{3!} \sum_{k=0}^{2} \frac{\phi_{k,0}(s)}{3!} \frac{\partial^3 K_{2m+1,2n+1}(x_k, \eta_2(t), u(x_k, \eta_2(t)))}{3!} \prod_{q=0}^{2} (t - y_{2j-1+q}).
\]

Similarly, for all \((s,t) \in [x_{2i-1}, x_{2i+1}] \times [c, y_1] \), there exists \((\xi_3(s), \eta_2(t)) \in [x_{2i-1}, x_{2i+1}] \times [c, y_1] \), such that

\[
R_3 = \frac{1}{3!} \frac{\partial^3 K_{2m+1,2n+1}(\xi_3(s), t, u(\xi_3(s), t))}{3!} \prod_{k=0}^{2} (s - x_{2i-1+k}) \\
+ \frac{2}{3!} \sum_{k=0}^{2} \frac{\phi_{k,2j-1}(s)}{3!} \frac{\partial^3 K_{2m+1,2n+1}(x_{2i-1+k}, \eta_2(t), u(x_{2i-1+k}, \eta_2(t)))}{3!} \prod_{q=0}^{2} (t - y_{2j-1+q}),
\]

and for all \((s,t) \in [x_{2i-1}, x_{2i+1}] \times [y_{2j-1}, y_{2j+1}] \), there exists \((\xi_4(s), \eta_2(t)) \in [x_{2i-1}, x_{2i+1}] \times [y_{2j-1}, y_{2j+1}] \), such that

\[
R_4 = \frac{1}{3!} \frac{\partial^3 K_{2m+1,2n+1}(\xi_4(s), t, u(\xi_4(s), t))}{3!} \prod_{k=0}^{2} (s - x_{2i-1+k}) \\
+ \frac{2}{3!} \sum_{k=0}^{2} \frac{\phi_{k,2j-1}(s)}{3!} \frac{\partial^3 K_{2m+1,2n+1}(x_{2i-1+k}, \eta_2(t), u(x_{2i-1+k}, \eta_2(t)))}{3!} \prod_{q=0}^{2} (t - y_{2j-1+q}).
\]
We can directly obtain \( r_{2m+1,2n+1}^{(1)} \) as follows

\[
| r_{2m+1,2n+1}^{(1)} | \leq \left| \int_a^{x_1} \int_c^{y_1} \frac{\partial^3 \kappa_{2m+1,2n+1}(\xi_1(s), t, u(\xi_1(s), t))}{3!} \sum_{k=0}^2 (s - x_k) dt ds \right|
\]

\[
+ \left| \int_a^{x_1} \int_c^{y_1} \frac{\sum_{k=0}^2 \varphi_{k,0}(s) \partial^3 \kappa_{2m+1,2n+1}(x_k, \eta_1(t), u(x_k, \eta_1(t)))}{3!} \sum_{q=0}^2 (t - y_q) dt ds \right|
\]

\[
\leq R_1^{(1)} + R_2^{(2)}. \tag{38}
\]

Next, we estimate each item on the right side of (38), and we have

\[
R_1^{(1)} \leq G_1 h_x^3 \left| \int_a^{x_1} \int_c^{y_1} (x_{2m+1} - s)^{-a} (y_{2n+1} - t)^{-\beta} dt ds \right|
\]

\[
= G_1 h_x^{4-a} h_y^{1-\beta} \left( \frac{(2m + 1)^{1-a} - (2n)^{1-a}}{(1-a)(1-\beta)} \right) \]

\[
\leq G_1 h_x^{4-a} h_y^{1-\beta}, \quad \forall \xi \in (2m, 2m + 1), \xi \in (2n, 2n + 1), \tag{39}
\]

\[
R_1^{(2)} \leq G_1 h_y^3 \left| \int_a^{x_1} \int_c^{y_1} (x_{2m+1} - s)^{-a} (y_{2n+1} - t)^{-\beta} dt ds \right| \leq G_1 h_x^{4-a} h_y^{4-\beta}, \tag{40}
\]

where \( G_1 \) is defined by (35).

Combining (39) and (40) yields

\[
| r_{2m+1,2n+1}^{(1)} | \leq G_1 (h_x^{4-a} h_y^{1-\beta} + h_x^{1-a} h_y^{4-\beta}). \tag{41}
\]

We use the same technique for \( r_{2m+1,2n+1}^{(2)} \), and it is easy to see that

\[
| r_{2m+1,2n+1}^{(2)} | \leq \sum_{j=1}^n \left| \int_a^{x_1} \int_{y_{2j-1}}^{y_{2j+1}} \frac{\partial^3 \kappa_{2m+1,2n+1}(\xi_2(s), t, u(\xi_2(s), t))}{3!} \sum_{k=0}^2 (s - x_k) dt ds \right|
\]

\[
+ \sum_{j=1}^n \left| \int_a^{x_1} \int_{y_{2j-1}}^{y_{2j+1}} \frac{\sum_{k=0}^2 \varphi_{k,0}(s) \partial^3 \kappa_{2m+1,2n+1}(x_k, \eta_2(t), u(x_k, \eta_2(t)))}{3!} \sum_{q=0}^2 (t - y_q) dt ds \right|
\]

\[
= R_1^{(1)} + R_2^{(2)}. \tag{42}
\]

For the first item on the right side of (42), we can obtain

\[
R_2^{(1)} \leq G_1 h_x^3 \sum_{j=1}^n \left| \int_a^{x_1} \int_{y_{2j-1}}^{y_{2j+1}} (x_{2m+1} - s)^{-a} (y_{2n+1} - t)^{-\beta} dt ds \right|
\]

\[
\leq G_1 h_x^{4-a} \sum_{j=1}^n \int_{y_{2j-1}}^{y_{2j+1}} (y_{2n+1} - t)^{-\beta} dt \leq G_1 h_x^{4-a} \int_{y_1}^{y_{2n+1}} (y_{2n+1} - t)^{-\beta} dt
\]

\[
= G_1 h_x^{4-a} \frac{(2n) h_x^{1-\beta}}{1-\beta} \leq \frac{d^{1-\beta}}{1-\beta} G_1 h_x^{4-a}. \tag{43}
\]
Next, we make a detailed estimate of $K_2^{(2)}$ with the following form

$$R_2^{(2)} \leq \sum_{j=1}^{n} \left| \int_{a}^{x_1} \int_{y_{j-1}}^{y_{j+1}} \frac{2}{k=0} \phi_{k,0}(s) \frac{1}{3!} \kappa_{2m+1,2n+1}(x_k, \eta_j(t), u(x_k, \eta_j(t))) \prod_{q=0}^{t-y_{j-1+q}} dt ds \right|$$

$$+ \sum_{j=1}^{n} \left| \int_{a}^{x_1} \int_{y_{j-1}}^{y_{j+1}} \frac{2}{k=0} \phi_{k,0}(s) \frac{1}{3!} \kappa_{2m+1,2n+1}(x_k, \eta_j(t), u(x_k, \eta_j(t))) \prod_{q=0}^{t-y_{j-1+q}} dt ds \right|$$

$$= A_1 + A_2,$$

(44)

where $\tilde{t}_j = y_{j+1}$. Based on the definition of $\phi_{k,i}(s), s \in (x_i, x_{i+2}), k = 0, 1, 2; i \in \mathbb{N}$, we know that $|\phi_{k,i}(s)| \leq 1$, and therefore $\sum_{k=0}^{2} |\phi_{k,i}(s)| \leq 3$, and we have

$$A_1 \leq G_1 \sum_{j=1}^{n} \left| \int_{a}^{x_1} \int_{y_{j-1}}^{y_{j+1}} \frac{2}{k=0} \phi_{k,0}(s) \prod_{q=0}^{t-y_{j-1+q}} dt ds \right|$$

$$\leq 3G_1h_x^{-\alpha} \sum_{j=1}^{n} \left| \int_{y_{j-1}}^{y_{j+1}} \frac{2}{q=0} \prod_{q=0}^{t-y_{j-1+q}} dt \right|$$

$$\leq 3G_1h_x^{-\alpha} \sum_{j=1}^{n} \left| \int_{y_{j-1}}^{y_{j+1}} \frac{2}{q=0} \prod_{q=0}^{t-y_{j-1+q}} dt \right|$$

$$+ 3G_1h_x^{-\alpha} \sum_{j=1}^{n} \left| \int_{y_{j-1}}^{y_{j+1}} \frac{2}{q=0} \prod_{q=0}^{t-y_{j-1+q}} dt \right|$$

$$= 3G_1h_x^{-\alpha} A_{11} + 3G_1h_x^{-\alpha} A_{12}.$$  

(45)

We make the following detailed estimates of $A_{11}$ and $A_{12}$, which can be directly obtained.

$$A_{11} \leq \sum_{j=1}^{n} \left| (y_{2n+1} - \tilde{t}_j)^{-\beta} \int_{y_{j-1}}^{y_{j+1}} \prod_{q=0}^{t-y_{j-1+q}} dt \right|$$

$$= \sum_{j=1}^{n} \left| \prod_{q=0}^{t-y_{j-1+q}} dt \right|$$

$$= 1^4 \sum_{j=1}^{n} \left| \prod_{q=0}^{t-y_{j-1+q}} dt \right|$$

$$= \frac{1}{4} h_y^{n-2} \sum_{j=1}^{n} \left| \prod_{q=0}^{t-y_{j-1+q}} dt \right|$$

$$= \frac{1}{4} h_y^{n-2} \sum_{j=1}^{n} \left| \prod_{q=0}^{t-y_{j-1+q}} dt \right|$$

$$= \frac{1}{4} h_y^{n-2} \sum_{j=1}^{n} \left| \prod_{q=0}^{t-y_{j-1+q}} dt \right|$$
$$\leq \frac{\hat{\beta}}{4} h_y^4 \int_{y_{2n-1}}^{y_{2n-1}} (y_{2n+1} - t)^{-\beta-1} dt = \frac{1}{4} h_y^4 \left( 2^{-\beta} h_y^- - (2n - 2)^{-\beta} h_y^- \right)$$

$$\leq \frac{2}{4} h_y^4 h_y^- = \frac{1}{2} h_y^4,$$

$$A_{12} \leq h_y^3 \left( \int_{y_{2n-3}}^{y_{2n-1}} (y_{2n+1} - t)^{-\beta} dt + \int_{y_{2n-1}}^{y_{2n-1}} (y_{2n+1} - t)^{-\beta} dt \right)$$

$$= h_y^3 \int_{y_{2n-3}}^{y_{2n-1}} (y_{2n+1} - t)^{-\beta} dt = \frac{4^{1-\beta}}{1-\beta} h_y^4,$$

(46)

(47)

where $\bar{t}_i \leq t_j \leq \bar{t}_i$, $y_{2j-1} \leq \bar{t}_i \leq y_{2j} \leq \bar{t}_i \leq y_{2j+1}$. Then, we insert (46) and (47) into (45) and obtain $A_1$ as follows

$$A_1 \leq 3 \left( \frac{1}{2} + \frac{4^{1-\beta}}{1-\beta} \right) G_1 h_x^{1-a} h_y^{4-\beta}.$$

(48)

For $A_2$, we discover

$$A_2 \leq G_2 h_y \int_a^{\infty} \left[ \sum_{j=0}^{n} \left( t - y_{2j-1} \right)^{\beta} \right] \frac{2}{q} \left( t - y_{2j-1+q} \right) dt \leq G_2 h_x^{1-a} h_y^{4-\beta} + \frac{G_2 d^{1-\beta}}{1-\beta} h_x^{1-a} h_y^{4-\beta}.$$

(49)

where $G_2$ is defined by (36).

According to (48) and (49), (44) becomes

$$R_2^{(2)} \leq 3 \left( \frac{1}{2} + \frac{4^{1-\beta}}{1-\beta} \right) G_1 h_x^{1-a} h_y^{4-\beta} + \frac{G_2 d^{1-\beta}}{1-\beta} h_x^{1-a} h_y^{4-\beta}.$$

(50)

Therefore, combining (43) and (50) with (42), we can obtain the following result:

$$|r_{2n+1,2n+1}^{(2)}| \leq \frac{d^{1-\beta}}{1-\beta} G_1 h_x^{4-\alpha} + \left[ 3 \left( \frac{1}{2} + \frac{4^{1-\beta}}{1-\beta} \right) G_1 h_y^{4-\beta} + \frac{G_2 d^{1-\beta}}{1-\beta} h_x^{1-a} h_y^{4-\beta} \right] h_x^{1-a}.$$

(51)

Now, we estimate $r_{2n+1,2n+1}^{(3)}$ and we can obtain

$$|r_{2n+1,2n+1}^{(3)}| \leq \sum_{i=1}^{m} \int_{x_{2i-1}}^{x_{2i-1}} \int_{c}^{y_{1}} \left[ \left( x_{2i+1} - s \right)^{\alpha} (y_{2n+1} - t)^{\beta} \right] \frac{2}{k} \left( s - x_{2i-1 + k} \right) dt ds$$

$$+ \sum_{i=1}^{m} \int_{x_{2i-1}}^{x_{2i-1}} \int_{c}^{y_{1}} \left[ \left( x_{2i+1} - s \right)^{\alpha - a} (y_{2n+1} - t)^{\beta} \right] \frac{2}{k} \left( s - x_{2i-1 + k} \right) dt ds$$

$$\leq R_{3}^{(1)} + R_{3}^{(2)}.$$
where $\delta_i = x_{2i}$. Taking the first term and the second term on the right side of the above formula as $B_1$ and $B_2$, we have

$$B_1 \leq G_1 h_y^{1-\beta} \sum_{i=1}^{m} \left| \int_{x_{2i-1}}^{x_{2i}} \frac{2}{k+1} \frac{(s-x_{2i-1+k})}{(x_{2m+1}-s)^{\alpha}} ds + \int_{x_{2i}}^{x_{2i+1}} \frac{2}{k+1} \frac{(s-x_{2i-1+k})}{(x_{2m+1}-s)^{\alpha}} ds \right|$$

$$\leq G_1 h_y^{1-\beta} \sum_{i=1}^{m-2} \left| \int_{x_{2i-1}}^{x_{2i}} \frac{2}{k+1} \frac{(s-x_{2i-1+k})}{(x_{2m+1}-s)^{\alpha}} ds + \int_{x_{2i}}^{x_{2i+1}} \frac{2}{k+1} \frac{(s-x_{2i-1+k})}{(x_{2m+1}-s)^{\alpha}} ds \right|$$

$$+ G_1 h_y^{1-\beta} \left( \int_{x_{2m-3}}^{x_{2m-1}} \frac{2}{k+1} \frac{(s-x_{2m-3+k})}{(x_{2m+1}-s)^{\alpha}} ds + \int_{x_{2m-1}}^{x_{2m+1}} \frac{2}{k+1} \frac{(s-x_{2m-1+k})}{(x_{2m+1}-s)^{\alpha}} ds \right)$$

$$\leq G_1 h_y^{1-\beta} B_{11} + G_1 h_y^{1-\beta} B_{12}.$$ 

For $B_{11}$ and $B_{12}$, we use the same method as (46) and (47), obtaining

$$B_{11} \leq \frac{1}{2} h_x^{4-\alpha}, \quad B_{12} \leq \frac{4^{1-\alpha}}{1-\alpha} h_x^{4-\alpha}.$$ 

Hence, for $B_1$, we can directly obtain

$$B_1 \leq \left( \frac{1}{2} + \frac{4^{1-\alpha}}{1-\alpha} \right) G_1 h_x^{4-\alpha} h_y^{1-\beta}.$$ 

For $B_2$, we can directly obtain the result

$$B_2 \leq G_2 h_x^4 \sum_{i=1}^{m} \int_{x_{2i-1}}^{x_{2i+1}} \int_{c}^{y_{2i}} \frac{(x_{2m+1}-s)^{-\alpha}}{(y_{2n+1}-t)^{\beta}} ds dt \leq \frac{b^{1-\alpha}}{1-\alpha} G_2 h_x^{4-\beta}.$$ 

Next, we estimate $R_{3}^{(2)}$ as follows

$$R_{3}^{(2)} \leq G_1 \sum_{i=1}^{m} \left| \int_{x_{2i-1}}^{x_{2i+1}} \frac{2}{k+1} \frac{\varphi(s)}{3(k\mu+1-s)} ds \right| \left| \int_{0}^{y_{2i}} \frac{1}{\mu(y_{2n+1}-t)^{\beta}} dt \right|$$

$$\leq G_1 h_y^{1-\beta} \sum_{i=1}^{m} \int_{x_{2i-1}}^{x_{2i+1}} \int_{c}^{y_{2i}} \frac{(x_{2m+1}-s)^{-\alpha}}{(y_{2n+1}-t)^{\beta}} ds dt \leq \frac{b^{1-\alpha}}{1-\alpha} G_1 h_y^{4-\beta}.$$ 

Therefore, we obtain that

$$|r_{2m+1,2n+1}^{(3)}| \leq \left( \frac{1}{2} + \frac{4^{1-\alpha}}{1-\alpha} \right) G_1 h_x^{4-\alpha} + \frac{b^{1-\alpha}}{1-\alpha} G_2 h_x^{4-\beta} + \frac{b^{1-\alpha}}{1-\alpha} G_1 h_y^{4-\beta}. \quad (52)$$

Finally, we make a detailed estimation of $r_{2m+1,2n+1}^{(4)}$, which has the form

$$|r_{2m+1,2n+1}^{(4)}| \leq \sum_{i=1}^{m} \sum_{j=1}^{n} \left| \int_{x_{2i-1}}^{x_{2i+1}} \int_{y_{2j-1}}^{y_{2j+1}} \frac{\partial^{3} \varphi_{2m+1,2n+1}(s,t)}{3! (x_{2m+1}-s)^{\alpha} (y_{2n+1}-t)^{\beta}} \prod_{k=0}^{2} (s-x_{2i-1+k}) ds dt \right|$$

$$+ \sum_{i=1}^{m} \sum_{j=1}^{n} \left| \int_{x_{2i-1}}^{x_{2i+1}} \int_{y_{2j-1}}^{y_{2j+1}} \frac{\varphi_{k,2i+1}(s)}{3! (y_{2n+1}-t)^{\beta}} \prod_{q=0}^{2} (t-y_{2j-1+q}) ds dt \right|$$

$$\times \partial^{3} \varphi_{2m+1,2n+1}(x_{2i-1+k}, y_{2j-1+q}, t).$$
\[ R_4^{(1)} = R_4^{(2)}. \]

Next, we estimate \( R_4^{(1)} \) and \( R_4^{(2)} \) one by one. For \( R_4^{(1)} \), we have

\[
R_4^{(1)} \leq \sum_{i=1}^{m} \sum_{j=1}^{n} \left| \int_{y_{2i-1}}^{y_{2i+1}} \int_{y_{2j-1}}^{y_{2j+1}} \frac{\partial_1^3 \kappa_{2m+1,2n+1}(\xi_{1i}(\tilde{s}_i), t, u(\xi_{1i}(\tilde{s}_i), t))}{3!(y_{2n+1} - t)^\beta} \right. \\
\left. \sum_{k=0}^{2} (s - x_{2i-1+k}) \right| \frac{d}{ds} \int_{y_{2i-1}}^{y_{2i+1}} \frac{(x_{2m+1} - s)^{-\alpha} \prod_{k=0}^{2} (s - x_{2i-1+k})}{(y_{2n+1} - t)^\beta} ds \\
+ \sum_{i=1}^{m} \sum_{j=1}^{n} \left| \int_{y_{2i-1}}^{y_{2i+1}} \int_{y_{2j-1}}^{y_{2j+1}} \frac{(x_{2m+1} - s)^{-\alpha} \prod_{k=0}^{2} (s - x_{2i-1+k})}{(y_{2n+1} - t)^\beta} \right| \frac{d}{ds} \int_{y_{2i-1}}^{y_{2i+1}} \frac{(x_{2m+1} - s)^{-\alpha} \prod_{k=0}^{2} (s - x_{2i-1+k})}{(y_{2n+1} - t)^\beta} ds \\
\times (\partial_2^3 \kappa_{2m+1,2n+1}(\xi_{1i}(s), t, u(\xi_{1i}(s), t)) - \partial_2^3 \kappa_{2m+1,2n+1}(\xi_{1i}(\tilde{s}_i), t, u(\xi_{1i}(\tilde{s}_i), t))) dt ds \right| \\
\leq P_1 + P_2,
\]

where \( \tilde{s}_i = x_{2i} \).

For \( P_1 \), the same processing method can be used as for \( B_1 \), to obtain

\[
P_1 \leq G_1 \sum_{i=1}^{m} \sum_{j=1}^{n} \left| \int_{y_{2i-1}}^{y_{2i+1}} \int_{y_{2j-1}}^{y_{2j+1}} \frac{(x_{2m+1} - s)^{-\alpha} \prod_{k=0}^{2} (s - x_{2i-1+k})}{(y_{2n+1} - t)^\beta} ds \right| \\
\leq d^{1-\beta} G_1 \sum_{i=1}^{m} \int_{y_{2i-1}}^{y_{2i+1}} \frac{d}{ds} \int_{y_{2i-1}}^{y_{2i+1}} \frac{(s - x_{2i-1+k})}{(x_{2m+1} - s)^\alpha} ds \\
\leq \frac{d^{1-\beta}}{1-\beta} \left( \frac{1}{2} + \frac{4^{1-\alpha}}{1-\alpha} \right) G_1 h_x^{4-\alpha}.
\]

For \( P_2 \), we can obtain

\[
P_2 \leq G_2 h_x^{4-\alpha} \sum_{i=1}^{m} \sum_{j=1}^{n} \int_{y_{2i-1}}^{y_{2i+1}} \int_{y_{2j-1}}^{y_{2j+1}} \frac{(x_{2m+1} - s)^{-\alpha}}{(y_{2n+1} - t)^\beta} ds dt ds \leq \frac{b^{1-\alpha} d^{1-\beta}}{(1-\alpha)(1-\beta)} G_2 h_x^{4-\alpha}.
\]

Finally, we estimate \( R_4^{(2)} \) as follows

\[
R_4^{(2)} \leq \sum_{i=1}^{m} \sum_{j=1}^{n} \int_{y_{2i-1}}^{y_{2i+1}} \int_{y_{2j-1}}^{y_{2j+1}} \frac{\partial_1^3 \kappa_{2m+1,2n+1}(x_{2i-1+k}, \eta_2(\tilde{t}_j), u(x_{2i-1+k}, \eta_2(\tilde{t}_j)) \prod_{j=0}^{2} (t - y_{2j-1+q}) dt ds}{3!(y_{2n+1} - t)^\beta} \\
\leq \sum_{i=1}^{m} \sum_{j=1}^{n} \int_{y_{2i-1}}^{y_{2i+1}} \int_{y_{2j-1}}^{y_{2j+1}} \frac{\partial_1^3 \kappa_{2m+1,2n+1}(x_{2i-1+k}, \eta_2(\tilde{t}_j), u(x_{2i-1+k}, \eta_2(\tilde{t}_j)) \prod_{j=0}^{2} (t - y_{2j-1+q}) dt ds}{3!(y_{2n+1} - t)^\beta} \\
- \frac{\partial_1^3 \kappa_{2m+1,2n+1}(x_{2i-1+k}, \eta_2(\tilde{t}_j), u(x_{2i-1+k}, \eta_2(\tilde{t}_j)) \prod_{j=0}^{2} (t - y_{2j-1+q}) dt ds}{3!(y_{2n+1} - t)^\beta} \\
+ \sum_{i=1}^{m} \sum_{j=1}^{n} \int_{y_{2i-1}}^{y_{2i+1}} \int_{y_{2j-1}}^{y_{2j+1}} \frac{\partial_2^3 \kappa_{2m+1,2n+1}(x_{2i-1+k}, \eta_2(\tilde{t}_j), u(x_{2i-1+k}, \eta_2(\tilde{t}_j)) \prod_{j=0}^{2} (t - y_{2j-1+q}) dt ds}{3!(y_{2n+1} - t)^\beta} \\
\times \frac{d}{ds} \left( \frac{\partial_2^3 \kappa_{2m+1,2n+1}(x_{2i-1+k}, \eta_2(\tilde{t}_j), u(x_{2i-1+k}, \eta_2(\tilde{t}_j)) \prod_{j=0}^{2} (t - y_{2j-1+q}) dt ds}{3!(y_{2n+1} - t)^\beta} \right) \\
\times \left( \frac{d}{ds} \int_{y_{2i-1}}^{y_{2i+1}} \frac{\partial_2^3 \kappa_{2m+1,2n+1}(x_{2i-1+k}, \eta_2(\tilde{t}_j), u(x_{2i-1+k}, \eta_2(\tilde{t}_j)) \prod_{j=0}^{2} (t - y_{2j-1+q}) dt ds}{3!(y_{2n+1} - t)^\beta} \right)
\]

where \( \tilde{t}_j = y_{2j} \). The two terms of the above formula are denoted \( J_1 \) and \( J_2 \), respectively. Using the same method as for \( A_1 \) to deal with \( J_1 \), we can obtain

\[
J_1 \leq G_1 \sum_{i=1}^{m} \int_{y_{2i-1}}^{y_{2i+1}} \left| \frac{\partial_2^3 \kappa_{2m+1,2n+1}(x_{2i-1+k}, \eta_2(\tilde{t}_j), u(x_{2i-1+k}, \eta_2(\tilde{t}_j)) \prod_{j=0}^{2} (t - y_{2j-1+q}) dt ds}{3!(y_{2n+1} - t)^\beta} \right| \\
\times \sum_{j=1}^{n} \int_{y_{2j-1}}^{y_{2j+1}} \frac{\partial_2^3 \kappa_{2m+1,2n+1}(x_{2i-1+k}, \eta_2(\tilde{t}_j), u(x_{2i-1+k}, \eta_2(\tilde{t}_j)) \prod_{j=0}^{2} (t - y_{2j-1+q}) dt ds}{3!(y_{2n+1} - t)^\beta} dt ds.
\]
Convergence Analysis

To simplify the notation, we rewrite the numerical scheme by introducing the following coefficients:

\[
\begin{align*}
\hat{B}_i & = \frac{\omega_{2m+1}^{(0)}}{h_x^{-a}} B_i, \quad i = 0, 1, 2; \\
B_i^n & = \frac{\omega_{2m+1}^{(0)}}{h_x^{-a}} B_i^n, \\
B_{2i+1}^m & = \frac{\omega_{2m+1}^{(0)}}{h_x^{-a}} B_{2i+1}^m, \\
\hat{D}_j & = \frac{\omega_{2m+1}^{(0)}}{h_y^{-\beta}} D_j, \quad j = 0, 1, 2; \\
D_j^n & = \frac{\omega_{2m+1}^{(0)}}{h_y^{-\beta}} D_j^n, \\
D_{2j+1}^m & = \frac{\omega_{2m+1}^{(0)}}{h_y^{-\beta}} D_{2j+1}^m, \\
\end{align*}
\]

Bringing (41) and (51)–(53) into (37) gives

\[
\begin{align*}
\frac{|r_{2m+1,2n+1}|}{G_1 h_x^{-a} + h_y^{-\beta}} & \leq G_1 \left[ \left( \frac{1}{2} + \frac{4^{1-a}}{1-a} \right) h_x^{-a} + \frac{4^{1-\beta}}{1-\beta} \frac{b^{1-a}}{1-a} h_y^{-\beta} \right] \\
& \quad + \frac{b^{1-a} d^{1-\beta} G_2}{(1-a)(1-\beta)} (h_x^4 + h_y^4),
\end{align*}
\]

Therefore, we obtain

\[
|r_{2m+1,2n+1}| \leq C(h_x^{-a} + h_y^{-\beta}),
\]

where \(C\) depends only on \(G_1, G_2, a\) and \(\beta\).

We can prove the truncation error at the other steps in a similar way to the truncation error at the point \((x_{2m+1}, y_{2n+1})\). We can draw the conclusion that the truncation errors \(r_{i,j}\) satisfy

\[
|r_{i,j}| \leq C(h_x^{-a} + h_y^{-\beta}), \quad i = 1, 2, \cdots, 2M; \quad j = 1, 2, \cdots, 2N.
\]

The proof is then completed. \(\square\)

4. Convergence Analysis

To simplify the notation, we rewrite the numerical scheme by introducing the following coefficients:

\[
\begin{align*}
\hat{B}_i & = \frac{\omega_{2m+1}^{(0)}}{h_x^{-a}} B_i, \quad i = 0, 1, 2; \\
B_i^n & = \frac{\omega_{2m+1}^{(0)}}{h_x^{-a}} B_i^n, \\
B_{2i+1}^m & = \frac{\omega_{2m+1}^{(0)}}{h_x^{-a}} B_{2i+1}^m, \\
\hat{D}_j & = \frac{\omega_{2m+1}^{(0)}}{h_y^{-\beta}} D_j, \quad j = 0, 1, 2; \\
D_j^n & = \frac{\omega_{2m+1}^{(0)}}{h_y^{-\beta}} D_j^n, \\
D_{2j+1}^m & = \frac{\omega_{2m+1}^{(0)}}{h_y^{-\beta}} D_{2j+1}^m, \\
\end{align*}
\]
Carefully observing (11), (15), (21) and (22), it is obvious that $A_{2m+1}^{k,i} = A_{2m+2}^{k,i}, k = 0, 1, 2, i = 1, 2, \ldots, m; A_{2n+1}^{q,j} = A_{2n+2}^{q,j}, q = 0, 1, 2, j = 1, 2, \ldots, n$. Hence, the numerical scheme (32) can be reformulated with an equivalent form as follows

\begin{align*}
    u_{1,1} &= f_{1,1} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{1,1}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2,1} &= f_{2,1} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2,1}(x_{i}, y_{j}, u_{i,j}), \\
    u_{1,2} &= f_{1,2} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{1,2}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2,2} &= f_{2,2} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2,2}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2m+1,1} &= f_{2m+1,1} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+1,1}(x_{i}, y_{j}, u_{i,j}) \\
    &+ h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=3}^{2m+1} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+1,1}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2m+2,1} &= f_{2m+2,1} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+2,1}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2m+1,2} &= f_{2m+1,2} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+1,2}(x_{i}, y_{j}, u_{i,j}) \\
    &+ h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=3}^{2m+1} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+1,2}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2m+2,2} &= f_{2m+2,2} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+2,2}(x_{i}, y_{j}, u_{i,j}), \\
    u_{1,2n+1} &= f_{1,2n+1} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{1,2n+1}(x_{i}, y_{j}, u_{i,j}) \\
    &+ h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=3}^{2n+1} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{1,2n+1}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2,2n+1} &= f_{2,2n+1} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2,2n+1}(x_{i}, y_{j}, u_{i,j}) \\
    &+ h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=3}^{2n+1} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2,2n+1}(x_{i}, y_{j}, u_{i,j}), \\
    u_{1,2n+2} &= f_{1,2n+2} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2n+2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{1,2n+2}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2,2n+2} &= f_{2,2n+2} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2n+2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2,2n+2}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2m+1,2n+1} &= f_{2m+1,2n+1} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+1,2n+1}(x_{i}, y_{j}, u_{i,j}) \\
    &+ h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=3}^{2m+1} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+1,2n+1}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2m+1,2n+2} &= f_{2m+1,2n+2} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+1,2n+2}(x_{i}, y_{j}, u_{i,j}), \\
    u_{2m+2,2n+1} &= f_{2m+2,2n+1} + h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+2,2n+1}(x_{i}, y_{j}, u_{i,j}) \\
    &+ h_x^{1-\alpha}h_y^{1-\beta} \sum_{i=3}^{2m+1} \sum_{j=0}^{2} B_{i}^{m} D_{j}^{n} \mathcal{K}_{2m+2,2n+1}(x_{i}, y_{j}, u_{i,j}).
\end{align*}
Let \( u \) be the exact solution of Equation (56), as in the following theorem.

**Proof.** Let \( u \) be the numerical solution of condition (2), and the step size \( h_x \) be small. It is easily seen that

\[
|u(x_i, y_j) - u_{i,j}| \leq C(h_x^{4 - \alpha} + h_y^{4 - \beta}), \quad i = 1, 2, \ldots, 2M; j = 1, 2, \ldots, 2N,
\]

where \( C \) depends only on \( L, \alpha, \beta, d, b, \) and \( \alpha \) and \( \beta \).

**Theorem 1.** Let \( u \) be the exact solution of Equation (1), \( u_{i,j}, i = 0, 1, \ldots, 2M, j = 0, 1, \ldots, 2N, \) be the numerical solution of (56). If \( \kappa, \kappa' \) satisfies the condition (2), and the step size \( h_x, h_y \) satisfies

\[
Lh_x^{4 - \alpha}h_y^{4 - \beta}|B_{2n+2}^n| |D_{2n+2}^n| < 1, 2Lh_x^{4 - \alpha}h_y^{4 - \beta}|B_{2n+2}^n| |D_{2n+2}^n| < 1, 2Lh_x^{4 - \alpha}h_y^{4 - \beta}|D_{2n+2}^n| < 1, CLh_x^{4 - \alpha}h_y^{4 - \beta} < 1,
\]

then the following error estimates hold

\[
|u(x_i, y_j) - u_{i,j}| \leq C(h_x^{4 - \alpha} + h_y^{4 - \beta}), \quad i = 1, 2, \ldots, 2M; j = 1, 2, \ldots, 2N,
\]

where \( C \) depends only on \( L, \alpha, \beta, d, b, \) and \( \alpha \) and \( \beta \).

**Proof.** Let \( e_{i,j} = u(x_i, y_j) - u_{i,j}, i = 0, 1, \ldots, 2M; j = 0, 1, \ldots, 2N. \) It is easily seen that \( e_{0,0} = e_{0,0} = 0. \) Firstly, \( e_{i,j}, i, j = 1, 2, \) satisfy

\[
e_{1,1} = h_x^{4 - \alpha}h_y^{4 - \beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_i^1 D_j^1 [\kappa_{1,1}(x_i, y_j, u(x_i, y_j)) - \kappa_{1,1}(x_i, y_j, u_{i,j})] + r_{1,1},
\]

\[
e_{1,2} = h_x^{4 - \alpha}h_y^{4 - \beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_i^1 D_j^1 [\kappa_{1,2}(x_i, y_j, u(x_i, y_j)) - \kappa_{1,2}(x_i, y_j, u_{i,j})] + r_{1,2},
\]

\[
e_{2,1} = h_x^{4 - \alpha}h_y^{4 - \beta} \sum_{i=0}^{2} \sum_{j=0}^{2} B_i^1 D_j^1 [\kappa_{2,1}(x_i, y_j, u(x_i, y_j)) - \kappa_{2,1}(x_i, y_j, u_{i,j})] + r_{2,1},
\]

and

\[
|e_{i,j}| \leq \sum_{i=0}^{2} \sum_{j=0}^{2} B_i^1 D_j^1 [\kappa_{i,j}(x_i, y_j, u(x_i, y_j)) - \kappa_{i,j}(x_i, y_j, u_{i,j})] + r_{i,j},
\]

where \( \kappa_{i,j} \) are functions depending on \( x, y, u, \) and \( \alpha, \beta, d, b, \) and \( \alpha \) and \( \beta \).
\[ e_{2,2} = h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} \tilde{B}_i \tilde{D}_j \left[ \kappa_{2,2}(x_i, y_j, u(x_i, y_j)) - \kappa_{2,2}(x_i, y_j, u_{ij}) \right] + r_{2,2}, \]

where \( \tilde{B}_i, \tilde{B}_j, \tilde{D}_i, \) and \( \tilde{D}_j \) are defined in (55). By direct calculation, it is easy to show that \( \tilde{B}_i, \tilde{B}_j, i = 0, 1, 2, \) and \( \tilde{D}_i, \tilde{D}_j, j = 0, 1, 2, \) are bounded and \( \kappa \) satisfies (2). Hence, it follows that

\[ |e_{1,1}| \leq CL h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} |\varepsilon_{ij}| + |r_{1,1}|, \quad |e_{1,2}| \leq CL h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} |\varepsilon_{ij}| + |r_{1,2}|, \]

\[ |e_{2,1}| \leq CL h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} |\varepsilon_{ij}| + |r_{2,1}|, \quad |e_{2,2}| \leq CL h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} |\varepsilon_{ij}| + |r_{2,2}|. \]

By combing these four inequalities, and taking into account (57), we discover that

\[ |\varepsilon_{ij}| \leq CL (|r_{1,1}| + |r_{1,2}| + |r_{2,1}| + |r_{2,2}|), \quad i, j = 1, 2. \]

Secondly, \( \varepsilon_{ij}, i \geq 3; j = 1, 2, \) satisfies

\[ e_{2m+1,1} = h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} B_i^m \tilde{D}_j \left[ \kappa_{2m+1,1}(x_i, y_j, u(x_i, y_j)) - \kappa_{2m+1,1}(x_i, y_j, u_{ij}) \right] \]

\[ + h_x^{1-a} h_y^{1-\beta} \sum_{i=3}^{2m+1} \sum_{j=0}^{2} B_i^m \tilde{D}_j \left[ \kappa_{2m+1,1}(x_{i-2}, y_j, u(x_{i-2}, y_j)) - \kappa_{2m+1,1}(x_{i-2}, y_j, u_{ij}) \right] + r_{2m+1,1}, \]

\[ e_{2m+2,1} = h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} B_i^m \tilde{D}_j \left[ \kappa_{2m+1,2}(x_i, y_j, u(x_i, y_j)) - \kappa_{2m+1,2}(x_i, y_j, u_{ij}) \right] \]

\[ + h_x^{1-a} h_y^{1-\beta} \sum_{i=3}^{2m+1} \sum_{j=0}^{2} B_i^m \tilde{D}_j \left[ \kappa_{2m+1,2}(x_{i-2}, y_j, u(x_{i-2}, y_j)) - \kappa_{2m+1,2}(x_{i-2}, y_j, u_{ij}) \right] + r_{2m+2,1}, \]

\[ e_{2m+1,2} = h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} B_i^m \tilde{D}_j \left[ \kappa_{2m+1,1}(x_i, y_j, u(x_i, y_j)) - \kappa_{2m+1,1}(x_i, y_j, u_{ij}) \right] \]

\[ + h_x^{1-a} h_y^{1-\beta} \sum_{i=3}^{2m+1} \sum_{j=0}^{2} B_i^m \tilde{D}_j \left[ \kappa_{2m+1,2}(x_i, y_j, u(x_i, y_j)) - \kappa_{2m+1,2}(x_i, y_j, u_{ij}) \right] + r_{2m+1,2}, \]

\[ e_{2m+2,2} = h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} B_i^m \tilde{D}_j \left[ \kappa_{2m+1,2}(x_i, y_j, u(x_i, y_j)) - \kappa_{2m+1,2}(x_i, y_j, u_{ij}) \right] + r_{2m+2,2}, \]

where \( B_i^m \) and \( B_j^m \) are in (55) and satisfy Lemma 2. Since the above four equations are coupled, they must be solved simultaneously. For simplicity, we choose to set \( ||\tilde{\varepsilon}_i|| = \max\{\varepsilon_{i,1}, \varepsilon_{i,2}, i = 0, 1, \ldots, 2M\} \) and \( ||\tilde{\varepsilon}_i|| = \max\{|r_{i,1}|, |r_{i,2}|, i = 0, 1, \ldots, 2M\} \), and then we can convert them into two groups of similar formulas, one of which is

\[ ||\tilde{\varepsilon}_{2m+1}|| \leq 2CL h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m} (2m + 2 - i)^{-a} ||\tilde{\varepsilon}_i|| + 2Lh_x^{1-a} h_y^{1-\beta} ||B_{2m+2}^a|| ||\tilde{\varepsilon}_{2m+1}|| + ||\tilde{\varepsilon}_{2m+1}||, \]

\[ ||\tilde{\varepsilon}_{2m+2}|| \leq 2CL h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m+1} (2m + 3 - i)^{-a} ||\tilde{\varepsilon}_i|| + 2Lh_x^{1-a} h_y^{1-\beta} ||B_{2m+2}^a|| ||\tilde{\varepsilon}_{2m+2}|| + ||\tilde{\varepsilon}_{2m+2}||. \]

For \( ||\tilde{\varepsilon}_{2m+1}|| \), we have

\[ ||\tilde{\varepsilon}_{2m+1}|| \leq 2CL h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m} (2m + 2 - i)^{-a} ||\tilde{\varepsilon}_i|| + C ||\tilde{\varepsilon}_{2m+1}|| \]

\[ \leq 2CL h_x^{1-a} h_y^{1-\beta} \sum_{i=0}^{2m} (2m + 1 - i)^{-a} ||\tilde{\varepsilon}_i|| + C ||\tilde{\varepsilon}_{2m+1}||. \]
After applying the Gronwall’s inequality [23] to the above equation, we obtain
\[
\|\tilde{e}_{2m+1}\| \leq C\|\tilde{e}_{2m+1}\| E_{1-a} ((2CLh_y^{1-\beta} \Gamma(1-a)((2m+1)h_x)^{1-a})
\leq C\|\tilde{e}_{2m+1}\| E_{1-a} (2CLh_y^{1-\beta} \Gamma(1-a)b^{1-a})
\]
Using the same method for \(\|\tilde{e}_{2m+2}\|\), we directly obtain
\[
\|\tilde{e}_{2m+2}\| \leq C\|\tilde{e}_{2m+2}\| E_{1-a} (2CLh_y^{1-\beta} \Gamma(1-a)b^{1-a})
\]

The process of calculating \(e_{i,j, i = 1, 2; j \geq 3}\), is similar to that for \(e_{i,j, i \geq 3; j = 1, 2}\), and therefore it is omitted.

Taking the above result, together with Lemma 1, we have
\[
\begin{align*}
|e_{i,j}| &\leq C(h_x^{1-a} + h_y^{4-\beta}), i \geq 1; j = 1, 2, \\
|e_{i,j}| &\leq C(h_x^{1-a} + h_y^{4-\beta}), i = 1, 2; j \geq 3.
\end{align*}
\]

Next, \(e_{i,j, i, j \geq 3}\) satisfy
\[
e_{2m+1,2n+1} = h_x^{1-a}y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} B_{i,j}^m D_{j}^n [k_{2m+1,2n+1}(x_i,y_j,u(x_i,y_j)) - k_{2m+1,2n+1}(x_i,y_j,u_{i,j})]
\]
\[
+ h_x^{1-a}y^{1-\beta} \sum_{i=0}^{2m+1} \sum_{j=0}^{2} B_{i,j}^m D_{j}^n [k_{2m+1,2n+1}(x_i,y_j,u(x_i,y_j)) - k_{2m+1,2n+1}(x_i,y_j,u_{i,j})]
\]
\[
+ h_x^{1-a}y^{1-\beta} \sum_{i=0}^{2m+1} \sum_{j=0}^{2} B_{i,j}^m D_{j}^n [k_{2m+1,2n+1}(x_i,y_j,u(x_i,y_j)) - k_{2m+1,2n+1}(x_i,y_j,u_{i,j})]
\]
\[
+ r_{2m+1,2n+1},
\]
\[
e_{2m+2,2n+1} = h_x^{1-a}y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} B_{i,j}^m D_{j}^n [k_{2m+2,2n+1}(x_i,y_j,u(x_i,y_j)) - k_{2m+2,2n+1}(x_i,y_j,u_{i,j})]
\]
\[
+ h_x^{1-a}y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2} B_{i,j}^m D_{j}^n [k_{2m+2,2n+1}(x_i,y_j,u(x_i,y_j)) - k_{2m+2,2n+1}(x_i,y_j,u_{i,j})]
\]
\[
+ r_{2m+2,2n+1},
\]
\[
e_{2m+1,2n+2} = h_x^{1-a}y^{1-\beta} \sum_{i=0}^{2m+1} \sum_{j=0}^{2} B_{i,j}^m D_{j}^n [k_{2m+1,2n+2}(x_i,y_j,u(x_i,y_j)) - k_{2m+1,2n+2}(x_i,y_j,u_{i,j})]
\]
\[
+ h_x^{1-a}y^{1-\beta} \sum_{i=0}^{2m+1} \sum_{j=0}^{2} B_{i,j}^m D_{j}^n [k_{2m+1,2n+2}(x_i,y_j,u(x_i,y_j)) - k_{2m+1,2n+2}(x_i,y_j,u_{i,j})]
\]
\[
+ r_{2m+1,2n+2},
\]
\[
e_{2m+2,2n+2} = h_x^{1-a}y^{1-\beta} \sum_{i=0}^{2m+1} \sum_{j=0}^{2} B_{i,j}^m D_{j}^n [k_{2m+2,2n+2}(x_i,y_j,u(x_i,y_j)) - k_{2m+2,2n+2}(x_i,y_j,u_{i,j})]
\]
\[
+ r_{2m+2,2n+2},
\]
where \(B_{i,j}^m, B_{i,j}^m, D_{j}^n\) and \(D_{j}^n\) are defined in (55) and satisfy Lemma 2. We estimate \(e_{2m+1,2n+1}\) as shown below.
\[
|e_{2m+1,2n+1}| \leq C(h_x^{1-a}y^{1-\beta} \sum_{i=0}^{2m+2} \sum_{j=0}^{2n} (2m + 2 - i)^{-a}(2n + 2 - j)^{-\beta}|e_{i,j}|
\]
which can be rearranged into the preferred form

\[
\begin{align*}
|e_{2m+1,2n+1}| & \leq CLh_x^{1-\beta} h_y^{1-\beta} \sum_{i=0}^{2m} (2m + 2 - i)^{-\beta} \sum_{j=0}^{2n} (2n + 2 - j)^{-\beta} |e_{i,j}| \\
& + CLh_x^{1-\beta} h_y^{1-\beta} \sum_{j=0}^{2n} (2n + 2 - j)^{-\beta} |e_{2m+1,j}| \\
& + CLh_x^{1-\beta} h_y^{1-\beta} \sum_{i=0}^{2m} (2m + 2 - i)^{-\beta} |e_{i,2n+1}| + C|r_{2m+1,2n+1}|.
\end{align*}
\]

If we let \( ||e_i|| = \max_{0 \leq j \leq 2N} |e_{i,j}| \), \( ||r_j|| = \max_{0 \leq j \leq 2N} |r_{i,j}| \), then we obtain

\[
\begin{align*}
|e_{2m+1,2n+1}| & \leq CLh_x^{1-\beta} h_y^{1-\beta} \sum_{i=0}^{2m} (2m + 2 - i)^{-\beta} \sum_{j=0}^{2n} ((2n + 2 - j)h_y)^{-\beta} ||e_i|| \\
& + CLh_x^{1-\beta} h_y^{1-\beta} \sum_{j=0}^{2n} ((2n + 2 - j)h_y)^{-\beta} ||e_{2m+1}|| \\
& + CLh_x^{1-\beta} h_y^{1-\beta} \sum_{i=0}^{2m} (2m + 2 - i)^{-\beta} ||e_{i,2n+1}|| + C||r_{2m+1}|| \\
& \leq CLh_x^{1-\beta} \sum_{i=0}^{2m} (2m + 2 - i)^{-\beta} ||e_i|| \sum_{j=0}^{2n} ((2n + 2 - j)h_y)^{-\beta} ||e_j|| \\
& + CLh_x^{1-\beta} ||e_{2m+1}|| \sum_{j=0}^{2n} ((2n + 2 - j)h_y)^{-\beta} dt \int_{y_0}^{y_{2n+2}} (y_{2n+2} - t)^{-\beta} dt \\
& + CLh_x^{1-\beta} ||e_{2m+1}|| \int_{y_0}^{y_{2n+2}} (y_{2n+2} - t)^{-\beta} dt \\
& + CLh_x^{1-\beta} ||e_{2m+1}|| \sum_{j=0}^{2n} (2m + 2 - i)^{-\beta} ||e_j|| + C||r_{2m+1}|| \\
& = CLh_x^{1-\beta} \frac{1}{1-\beta} \sum_{i=0}^{2m} (2m + 2 - i)^{-\beta} ||e_i|| + CLh_x^{1-\beta} \frac{1}{1-\beta} ||e_{2m+1}|| \\
& + CLh_x^{1-\beta} h_y^{1-\beta} \sum_{j=0}^{2n} (2m + 2 - i)^{-\beta} ||e_j|| + C||r_{2m+1}|| \\
& \leq CLh_x^{1-\beta} \frac{d^{1-\beta}}{1-\beta} \sum_{i=0}^{2m} (2m + 2 - i)^{-\beta} ||e_i|| + CLh_x^{1-\beta} \frac{d^{1-\beta}}{1-\beta} ||e_{2m+1}|| \\
& + CLh_x^{1-\beta} h_y^{1-\beta} \sum_{j=0}^{2n} (2m + 2 - i)^{-\beta} ||e_j|| + C||r_{2m+1}||.
\end{align*}
\]
The corresponding exact solution is $u(x, y) = x^4 y^4$.

5. Numerical Examples

In this section, the numerical scheme is used to solve the two-dimensional fractional Volterra integral equations, and we propose four calculation examples to prove its effectiveness. All the numerical examples were implemented using MATLAB on a ThinkCentre computer with a 3.40 GHz Intel(R) Core(TM) i5-7500 CPU and 8.00 GB of RAM. The time complexity of the proposed numerical scheme on discrete grids was $O((MN)^2)$.

Example 1. Consider the following two-dimensional linear fractional Volterra integral equations

$$u(x, y) = f(x, y) + \int_0^x \int_0^y \frac{(xy + s + 1)u(s, t)}{(x - s)^\alpha (y - t)^\beta} \, dt \, ds, \quad (x, y) \in [0, 1] \times [0, 1],$$

where

$$f(x, y) = x^4 y^4 - 576 x^{6-\alpha} y^{6-\beta} \frac{\prod_{i=1}^5 \frac{1}{(i-\alpha)}}{\prod_{j=1}^5 \frac{1}{(j-\beta)}} - 2880 x^{6-\alpha} y^{6-\beta} \frac{\prod_{i=1}^6 \frac{1}{(i-\alpha)}}{\prod_{j=1}^6 \frac{1}{(j-\beta)}} = \frac{1}{\Gamma(\alpha+\beta+2)}.$$
In our test, we chose $\alpha = 0.3, \beta = 0.6$ and $\alpha = 0.1, \beta = 0.7$, respectively. The step size in the x-direction was $h_x = \frac{1}{2N}$ and the step size in the y-direction was $h_y = \frac{1}{N}$. The definition of error was as follows

$$e^u_h = \max_{i=1,\ldots,2M} \max_{j=1,\ldots,2N} |u(x_i, y_j) - u_{i,j}|.$$ 

In this example, we wished to check how well the constructed numerical format converged for different choices of $\alpha$ and $\beta$. The test results are given in Tables 1 and 2, and the convergence order was calculated as $\log_{10} \left( \frac{e_{h1}}{e_{h2}} \right)$. From the results of the theoretical analysis in Theorem 1, we know that the theoretical convergence order of the constructed numerical scheme is $O(h_x^{4-\alpha} + h_y^{4-\beta})$. When $h_x$ is sufficiently small with respect to $h_y$, we have $O(h_x^{4-\alpha} + h_y^{4-\beta}) = O(h_y^{4-\beta})$. In Tables 1 and 2, we take $\alpha = 0.3, \beta = 0.6$ or $\alpha = 0.1, \beta = 0.7$ and $M = 2N$. In this case, our scheme’s theoretical order is 4. In Tables 1 and 2, we show the corresponding convergence order and CPU time when $N$ takes a series of values and $\alpha$ and $\beta$ are given specific values. In Table 1, when $\alpha = 0.3, \beta = 0.6$, the tested order is close to the theoretical order of 3.2. In Table 2, when $\alpha = 0.1, \beta = 0.7$, the order of the test is close to the theoretical order of 3.3. This is consistent with the theoretical prediction. However, from Tables 1 and 2, we find that the CPU time quickly increases from 0.14 s to about 2 h as $N$ increases.

**Table 1.** Maximum errors, decay rate and CPU time with $M = 2N$.

| $N$  | $\alpha = 0.3, \beta = 0.6$ | Order | CPU Time          |
|------|-----------------------------|-------|------------------|
| 8    | $1.9973659945 \times 10^{-3}$ | -     | 0.1438033000 s   |
| 16   | $2.0439893391 \times 10^{-4}$ | 3.2886391376 | 1.779478000 s     |
| 32   | $2.0217387802 \times 10^{-5}$ | 3.3377191614 | 27.3948994000 s   |
| 64   | $1.9623458836 \times 10^{-6}$ | 3.3649453461 | 7.1728836283 min  |
| 128  | $1.8839724669 \times 10^{-7}$ | 3.3807295678 | 1.8956612181 h    |

**Table 2.** Maximum errors, decay rate and CPU time with $M = 2N$.

| $N$  | $\alpha = 0.1, \beta = 0.7$ | Order | CPU Time          |
|------|-----------------------------|-------|------------------|
| 8    | $1.0362949128 \times 10^{-3}$ | -     | 0.2546744000 s   |
| 16   | $1.1195268795 \times 10^{-4}$ | 3.2104735669 | 1.827023000 s     |
| 32   | $1.1762915084 \times 10^{-5}$ | 3.2505716291 | 27.406621300 s   |
| 64   | $1.2176430484 \times 10^{-6}$ | 3.2720824624 | 7.1583191750 min  |
| 128  | $1.2502313451 \times 10^{-7}$ | 3.2838242818 | 1.891908256 h    |

Similarly, we test the convergence order using the proposed numerical scheme in another way. When we take $N = \ceil{M^{4-\beta}}$, where $\cdot$ indicates rounding up, the numerical scheme’s theoretical order $O(h_x^{4-\alpha} + h_y^{4-\beta}) = O(h_x^{4-\alpha})$. The test results are given in Table 3. When $\alpha = 0.3, \beta = 0.6$, the numerical result of the test is close to 3.7, and when $\alpha = 0.1, \beta = 0.7$, the numerical result of the test is close to 3.9. It may be noted from Tables 1–3 that the high-order numerical scheme is convergent and has good approximation.

**Table 3.** Maximum errors and decay rate with $N = \ceil{M^{(4-\alpha)/(4-\beta)}}$.

| $M$  | $\alpha = 0.3, \beta = 0.6$ | Order | $\alpha = 0.1, \beta = 0.7$ | Order   |
|------|-----------------------------|-------|-----------------------------|---------|
| 8    | $2.7093547741 \times 10^{-3}$ | -     | $8.5837427821 \times 10^{-4}$ | -       |
| 16   | $2.4017036964 \times 10^{-4}$ | 3.4958192397 | $6.788704124 \times 10^{-5}$ | 3.6603986275 |
| 32   | $2.0350262447 \times 10^{-5}$ | 3.5609388681 | $5.0238828559 \times 10^{-6}$ | 3.7562615829 |
| 64   | $1.6641178999 \times 10^{-6}$ | 3.6122178456 | $3.630382318 \times 10^{-7}$ | 3.790609690 |
| 128  | $1.336559233 \times 10^{-7}$ | 3.6380630008 | $2.5643266843 \times 10^{-8}$ | 3.8234698786 |

**Example 2.** Consider the following two-dimensional nonlinear fractional Volterra integral equations:
Fractal Fract. 2022, 6, 314

u(x, y) = f(x, y) + \int_0^x \int_0^y (xy + s + t) u^2(s, t) \frac{dtds}{(x-s)^\alpha (y-t)^\beta}, \quad (x, y) \in [0,1] \times [0,1],

where

f(x, y) = x^3 y^2 - 17280 x^{8-\alpha} y^{6-\beta} \prod_{i=1}^{7} \frac{1}{(i-\alpha)} \prod_{j=1}^{5} \frac{1}{(j-\beta)}

- 120960 x^{8-\alpha} y^{5-\beta} \prod_{i=1}^{8} \frac{1}{(i-\alpha)} \prod_{j=1}^{5} \frac{1}{(j-\beta)} - 86400 x^{7-\alpha} y^{6-\beta} \prod_{i=1}^{7} \frac{1}{(i-\alpha)} \prod_{j=1}^{6} \frac{1}{(j-\beta)},

and the exact solution is \( u(x, y) = x^3 y^2 \).

In this example, the meanings described in Tables 4 and 5 are similar to those in Tables 1–3 in Example 1. In Table 4, we can easily see that when \( \alpha = 0.3, \beta = 0.6 \) and \( \alpha = 0.1, \beta = 0.7 \), the numerical results of the test are very close to the theoretical values of 3.4 and 3.3, respectively. In Table 5, when \( \alpha = 0.3, \beta = 0.6 \), the numerical result of the test is close to 3.7, and when \( \alpha = 0.1, \beta = 0.7 \), the numerical result of the test is close to 3.9. This shows very good consistency with the theoretical prediction.

**Table 4.** Maximum errors and decay rate with \( M = 2N \).

| \( N \) | \( \alpha = 0.3, \beta = 0.6 \) | Order | \( \alpha = 0.1, \beta = 0.7 \) | Order |
|------|----------------|-------|----------------|-------|
| 8    | 1.4497410528 x 10^{-3} | –      | 6.7268209314 x 10^{-4} | –      |
| 16   | 1.4126072922 x 10^{-4} | 3.3593628803 | 6.9945531354 x 10^{-5} | 3.2656105874 |
| 32   | 1.3572271646 x 10^{-5} | 3.3796320927 | 7.1896243538 x 10^{-6} | 3.2822435908 |
| 64   | 1.2887160350 x 10^{-6} | 3.3966501211 | 7.3370971254 x 10^{-7} | 3.2926351059 |
| 128  | 1.2153830342 x 10^{-7} | 3.4064514616 | 7.4615217116 x 10^{-8} | 3.297667925 |

**Table 5.** Maximum errors and decay rate with \( N = \lceil M^{(4-\alpha)/(4-\beta)} \rceil \).

| \( M \) | \( \alpha = 0.3, \beta = 0.6 \) | Order | \( \alpha = 0.1, \beta = 0.7 \) | Order |
|------|----------------|-------|----------------|-------|
| 8    | 4.1299227631 x 10^{-3} | –      | 1.2444932111 x 10^{-3} | –      |
| 16   | 3.5190750852 x 10^{-4} | 3.5528465991 | 9.1544255179 x 10^{-5} | 3.7649451968 |
| 32   | 2.967317197 x 10^{-5} | 3.5677637121 | 6.6471665752 x 10^{-6} | 3.7836579403 |
| 64   | 2.4390547260 x 10^{-6} | 3.6049666456 | 4.7740052334 x 10^{-7} | 3.7994674546 |
| 128  | 1.9706716024 x 10^{-7} | 3.6295628422 | 3.3836403013 x 10^{-8} | 3.8185520322 |

Next, we plot the error distribution. The error distribution of \( M = 2N, N = 128 \) is shown in Figure 1, where the error distribution of \( \alpha = 0.3, \beta = 0.6 \) is on the left and the error distribution of \( \alpha = 0.1, \beta = 0.7 \) is on the right. From Figure 1, we find that the errors can be as small as \( 10^{-7} \) and \( 10^{-8} \).

![Figure 1. Error distribution of \( \alpha = 0.3, \beta = 0.6 \) (left) and \( \alpha = 0.1, \beta = 0.7 \) (right) for \( N = 128 \).](image_url)
Example 3. Assume the below two-dimensional fractional Volterra integral equation [24]:

\[
\begin{align*}
  u(x,y) &= x^2(y^2 - y) - 0.0828x^2y(3y - 4) \\
  &+ \frac{1}{\Gamma\left(\frac{\alpha}{2}\right)\Gamma\left(\frac{\beta}{2}\right)} \int_0^x \int_0^y (x-s)^{\frac{\alpha}{2}}(y-t)^{\frac{\beta}{2}} u(s,t)dsdt,
\end{align*}
\]

where the exact solution is \(u(x,y) = x^2(y^2 - y)\).

In this example, we take \(M = N = 32\) and compare our scheme with the numerical method in [24]. The numerical solutions of different points \((x, y) \in [0, 1] \times [0, 1]\) are given in Table 6, where “Haar solution” denotes the numerical solutions in Table 2 of [24], and “Numerical solution” denotes the numerical solutions obtained by our method. Since we are using a quadratic polynomial, while a zero polynomial is used in [24], combined with the data in Table 6, we can clearly see that the error produced by using our proposed method is much smaller.

Table 6. The numerical results in Example 3.

| \((x, y)\)   | Exact Solution | Haar Solution in [24] | Numerical Solution |
|-------------|----------------|-----------------------|--------------------|
| (0, 0)      | 0              | -0.006901             | -0.00089999999999999 |
| (0.1, 0.1)  | -0.003900     | -0.006458             | -0.00640000000000000 |
| (0.2, 0.2)  | -0.106200     | -0.062504             | -0.06250000000000000 |
| (0.3, 0.3)  | -0.109600     | -0.086475             | -0.086399999999997 |
| (0.4, 0.4)  | -0.083800     | -0.038412             | -0.03840000000000000 |
| (0.5, 0.5)  | -0.062500     | -0.018932             | -0.01899000000000000 |
| (0.6, 0.6)  | -0.062500     | -0.006458             | -0.00664000000000000 |
| (0.7, 0.7)  | -0.062500     | -0.006458             | -0.00664000000000000 |
| (0.8, 0.8)  | -0.062500     | -0.006458             | -0.00664000000000000 |
| (0.9, 0.9)  | -0.062500     | -0.006458             | -0.00664000000000000 |
| (1, 1)      | 0              | 0                     | 0.000000000000000137 |

Example 4. We consider the following two-dimensional nonlinear fractional Volterra integral equation [25]:

\[
\begin{align*}
  u(x,y) &= \sqrt{y} \left( -\frac{1}{180} x^2 y^3 \right) + \sqrt{\frac{x}{3}} \\
  &+ \frac{1}{\Gamma\left(\frac{\alpha}{2}\right)\Gamma\left(\frac{\beta}{2}\right)} \int_0^x \int_0^y (x-s)^{\frac{\alpha}{2}}(y-t)^{\frac{\beta}{2}} \sqrt{\frac{x}{yt}} u(s,t)dsdt,
\end{align*}
\]

with the exact solution \(u(x,y) = \sqrt{\frac{3xy}{5}}\).

In this example, we take \(M = N = 8, 32\) and give the exact and approximate solutions at the point \((x, y) \in [0, 1] \times [0, 1]\) in Table 7, where “Method solution” represents the approximate solution in Table 1 of [25] and “Numerical solution” represents the numerical solution using our proposed method. The maximum error is also given in Table 7. According to the calculation results shown in Table 7, we can see that the approximate solution obtained using our proposed method is closer to the exact solution at the corresponding point, and the maximum error is smaller.

Table 7. The numerical results for Example 4.

| \(x = y\) | Exact Solution | Method Solution in [25] | Numerical Solution |
|-----------|----------------|------------------------|--------------------|
|           |                | \(h = 1/16\) | \(h = 1/64\) | \(h = 1/16\) | \(h = 1/64\) |
| 0         | 0              | 0.018452              | 0.001216           | 0              | 0              |
| 0.1       | 0.057735       | 0.031133              | 0.006458           | 0.062761       | 0.057660       |
| 0.2       | 0.11547        | 0.13261               | 0.117302           | 0.114850       | 0.115427       |
| 0.3       | 0.133525       | 0.14845               | 0.168105           | 0.171241       | 0.173187       |
| 0.4       | 0.25694        | 0.246768              | 0.25442            | 0.258408       | 0.258358       |
| 0.5       | 0.286875       | 0.262075              | 0.265335           | 0.288662       | 0.288662       |
| 0.6       | 0.34641        | 0.36925               | 0.347581           | 0.346202       | 0.346324       |
| 0.7       | 0.401415       | 0.37535               | 0.39155            | 0.404047       | 0.404057       |
| 0.8       | 0.46188        | 0.475083              | 0.462721           | 0.461606       | 0.461675       |
| 0.9       | 0.519615       | 0.501015              | 0.513255           | 0.519441       | 0.519463       |
| 0.99      | 0.571577       | 0.583533              | 0.572104           | 0.570976       | 0.571549       |
| Max error | 0              | 2.96 \times 10^{-2}    | 8.69 \times 10^{-3} | 5.02 \times 10^{-3} | 1.51 \times 10^{-4} |

Fractal Fract. 2022, 6, 314
6. Concluding Remarks

Following the modified block-by-block method [5], we proposed an efficient and high-order numerical scheme (32) to approximate the solution of two-dimensional nonlinear fractional Volterra integral equations with singular kernels. The main idea of the high-order numerical scheme was to discretize its domain into a number of subdomains and then use biquadratic interpolation on each subdomain. In this high-order numerical scheme, only the two boundary layers were coupled, and the rest of the blocks were explicit in the scheme, which made our calculations more convenient. The scheme has uniform accuracy, and the optimal convergence order was $O(h^4_x + h^4_y)$. For the high-order numerical scheme, we conducted a detailed error analysis and verified the correctness of the theoretical analysis through numerical examples. The advantages of the numerical scheme (32) are high accuracy, easy calculation and the fact that there is no need for coupling solutions. The disadvantage of the numerical scheme is that the computation time is long, increasing as the number of partitions increases. In the future, according to reference [26,27], we intend to use a fast algorithm to realize the results. Based on the idea of [28], we will use the high-order numerical scheme to solve Fredholm–Hammerstein integral equations of the second kind. Based on the ideas of [29–32], we expect that the constructed efficient higher-order scheme can be applied to engineering and practical problems.

Author Contributions: Funding acquisition, Z.-Q.W. and J.-Y.C.; investigation, Z.-Q.W. and Q.L.; methodology, Z.-Q.W. and J.-Y.C.; project administration, Z.-Q.W. and J.-Y.C.; software, Q.L.; supervision, Z.-Q.W. and J.-Y.C.; visualization, Q.L.; writing—original draft, Z.-Q.W. and Q.L.; writing—review and editing, Z.-Q.W., Q.L. and J.-Y.C.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by National Natural Science Foundation of China (Grant Nos. 11961009 and 11901135) and the Foundation of Guizhou Science and Technology Department (Grant No. [2020]1Y015).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All the data were computed using our high-order numerical scheme.

Acknowledgments: The authors thank the anonymous referees for their valuable suggestions, which improved the quality of this work significantly.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study, in the collection, analyses or interpretation of data, in the writing of the manuscript or in the decision to publish the results.

References
1. Kimeu, J. Fractional Calculus: Definitions and Applications. Master’s Thesis, Western Kentucky University, Bowling Green, Kentucky, USA, 2009.
2. Micula, S. A numerical method for weakly singular nonlinear Volterra integral equations of the second kind. *Symmetry* 2020, 12, 1862.
3. Saberi-Nadjafi, J.; Heidari, M. A generalized block-by-block method for solving linear Volterra integral equations. *Appl. Math. Comput.* 2007, 188, 1969–1974. [CrossRef]
4. Katani, R.; Shahmorad, S. Block by block method for the systems of nonlinear Volterra integral equations. *Appl. Math. Model.* 2010, 34, 400–406. [CrossRef]
5. Cao, J.; Xu, C. A high order schema for the numerical solution of the fractional ordinary differential equations. *J. Comput. Phys.* 2013, 238, 154–168. [CrossRef]
6. Assari, P.; Dehghan, M. The approximate solution of nonlinear Volterra integral equations of the second kind using radial basis functions. *Appl. Numer. Math.* 2018, 131, 140–157. [CrossRef]
7. Al-Bugami, A.M.; Al-Juaid, J.G. Runge-Kutta method and block by block method to solve nonlinear Fredholm-Volterra integral equation with continuous Kernel. *J. Appl. Math. Phys.* 2020, 8, 2043–2054. [CrossRef]
8. Jaabar, S.M.; Hussain, A.H. Solving Volterra integral equation by using a new transformation. *J. Interdiscip. Math.* 2021, 24, 735–741. [CrossRef]
9. Usta, F. Bernstein approximation technique for numerical solution of Volterra integral equations of the third kind. *Comput. Appl. Math.* 2021, 40, 161–172.
10. Ma, Z.; Alikhanov, A.A.; Huang, C.; Zhang, G. A multi-domain spectral collocation method for Volterra integral equations with a weakly singular kernel. *Appl. Numer. Math.* 2021, 167, 218–236. [CrossRef]
11. Mckee, S.; Tang, T.; Diogo, T. An Euler-type method for two-dimensional Volterra integral equations of the first kind. *IMA J. Numer. Anal.* 2020, 20, 423–440. [CrossRef]
12. Wang, Y.; Huang, J.; Wen, X. Two-dimensional Euler polynomials solutions of two-dimensional Volterra integral equations of fractional order. *Appl. Numer. Math.* 2021, 163, 77–95. [CrossRef]
13. Abdi, A.; Conte, D. Implementation of general linear methods for Volterra integral equations. *J. Comput. Appl. Math.* 2021, 386, 1–19. [CrossRef]
14. Dehbozorgi, R.; Nedaiaasl, K. Numerical solution of nonlinear weakly singular Volterra integral equations of the first kind: An hp-version collocation approach. *Appl. Numer. Math.* 2021, 161, 111–136. [CrossRef]
15. Ahsan, S.; Nawaz, R.; Akbar, M.; Nisar, K.S.; Baleanu, D. Approximate solutions of nonlinear two-dimensional Volterra integral equations. *Math. Methods Appl. Sci.* 2021, 44, 5548–5559. [CrossRef]
16. Kant, K.; Nelakanti, G. Approximation methods for second kind weakly singular Volterra integral equations. *J. Comput. Appl. Math.* 2020, 368, 112531.
17. Pan, Y.; Huang, J.; Ma, Y. Bernstein series solutions of multidimensional linear and nonlinear Volterra integral equations with fractional order weakly singular kernels. *Appl. Math. Comput.* 2019, 347, 149–161. [CrossRef]
18. Allaei, S.S.; Diogo, T.; Rebelo, M. The Jacobi collocation method for a class of nonlinear Volterra integral equations with weakly singular kernel. *J. Sci. Comput.* 2016, 69, 673–695.
19. Li, X.; Tang, T.; Xu, C. Numerical solutions for weakly singular Volterra integral equations using Chebyshev and Legendre Pseudo-Spectral Galerkin methods. *J. Sci. Comput.* 2016, 67, 43–64. [CrossRef]
20. Abdulrahman, M.; Hassan, S.; Alomair, R.; Alsaleh, D. Fundamental solutions for the conformable time fractional Phi-4 and space-time fractional simplified MCH equations. *AIMS Math.* 2021, 6, 6555–6568. [CrossRef]
21. Abdulrahman, M.; Sohaly, M.; Alharbi, Y. Fundamental stochastic solutions for the conformable fractional NLSE with spatiotemporal dispersion via exponential distribution. *Phys. Scr.* 2021, 96, 125223. [CrossRef]
22. Wang, K.; Wang, G.; Zhu, H. A new perspective on the study of the fractal coupled Boussinesq-Burger equation in shallow water. *Fractals* 2021, 29, 2150122. [CrossRef]
23. Dixon, J.; Mckee, S. Weakly singular discrete Gronwall inequalities. *Z. Angew. Math. Mech.* 1978, 66, 535–544. [CrossRef]
24. Abdollahi, Z.; Moghadam, M.M.; Saeedi, H.; Ebadi, M. A computational approach for solving fractional Volterra integral equations based on two-dimensional Haar wavelet method. *Int. J. Comput. Math.* 2022, 99, 1488–1504.
25. Najafalizadeh, S.; Ezzati, R. Numerical methods for solving two-dimensional nonlinear integral equations of fractional order by using two-dimensional block pulse operational matrix. *Appl. Math. Comput.* 2016, 280, 46–56. [CrossRef]
26. Micula, S. A fast converging iterative method for Volterra integral equations of the second kind with delayed arguments. *Fixed Point Theory Appl.* 2015, 16, 371–380.
27. Gu, X.; Wu, S. A parallel-in-time iterative algorithm for Volterra partial integro-differential problems with weakly singular kernel. *J. Comput. Phys.* 2020, 417, 109576. [CrossRef]
28. Micula, S.; Cattani, C. On a numerical method based on wavelets for Fredholm-Hammerstein integral equations of the second kind. *Math. Methods Appl. Sci.* 2018, 41, 9103–9115.
29. Cao, J.; Cai, Z. Numerical analysis of a high-order scheme for nonlinear fractional differential equations with uniform accuracy. *Numer. Math. Theor. Meth. Appl.* 2014, 14, 71–112. [CrossRef]
30. Wang, Z.; Cui, J. Second-order two-scale method for bending behavior analysis of composite plate with 3-D periodic configuration and its approximation. *Sci. China Math.* 2014, 57, 1713–1732. [CrossRef]
31. Huang, Y.; Gu, X.; Gong, Y.; Li, H.; Zhao, Y.; Carpentieri, B. A fast preconditioned semi-implicit difference scheme for strongly nonlinear space-fractional diffusion equations. *Fract. Fract.* 2021, 5, 230. [CrossRef]
32. Gu, X.; Sun, H.; Zhao, Y.; Zheng, X. An implicit difference scheme for time-fractional diffusion equations with a time-invariant type variable order. *Appl. Math. Lett.* 2021, 120, 107270. [CrossRef]