Research Paper
Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. *, No. *, *, pp. 1-13
DOI:10.22034/cmde.2024.60254.2570

Numerical solutions and error analysis of system of two-dimensional Volterra integral equations via Block-Pulse functions

Akram Karimi, Khosrow Maleknejad*, and Reza Ezzati
Department of Mathematics, Karaj Branch, Islamic Azad University, Karaj, Iran.

Abstract

This paper tries to provide an attractive framework based on Block-Pulse functions for numerical solution of a system of two-dimensional Volterra integral equations of the second kind. These types of systems are created through the modeling of physics or engineering phenomena. By constructing operational matrices based on Block-Pulse functions and reduction of variables, a simpler algorithm is built. The block-pulse method is affordable because it converts algebraic systems to a matrix system and reduces the amount of computation. Some numerical examples and error analysis which are in detail support the method.

Keywords. Error analysis, Two-dimensional Block-Pulse function, System of integral equation, Operational matrix.
2010 Mathematics Subject Classification. 45F05.

1. Introduction

The system of Volterra integral equations (2D-VIEs) has a main niche in applied fields like engineering, physics, etc. For more details, see [? ? ?]. In the analysis of some classes of differential-algebraic systems of PDE [?] or in the modeling of certain heat conduction processes, the system of 2D-VIEs may arise. Here, with the help of two-dimensional Block-Pulse functions (2D-BPfs), we find the solution of a system of 2D-VIE as:

$$\sum_{j=1}^{l}\lambda_{ij}u_j(x,y)=g_i(x,y)+\sum_{j=1}^{l}\int_0^y\int_0^x k_{ij}(x,y,s,t)u_j(s,t)dsdt, \tag{1.1}$$

$i=1,2,...,l,$

$g_i(x,y)$ is known and $u_j(x,y)$ is unknown functions, respectively, defined on $[0,x]\times[0,y]$. The kernel can be separable or not; both types are solved in examples.

λ_{ij}, $k_{ij}(x,y,s,t)$ can be linear functions or even constants. The matrix λ_{ij} does not degenerate at any point in the integration domain, then this is the system of integral equations of the second kind. The uniqueness of the solution of system of 2D-VIE has been given in [? ?]. Through the paper, suppose that:

$$\mathbf{u}=[u_1(x,y),\cdots,u_l(x,y)]^T. \tag{1.2}$$

We want to get solution using the procedure as:

$$\mathbf{u}\simeq C\Psi, \tag{1.3}$$

that we try to find C, and, Ψ is introduced later. Also, we will present more details about $\mathbf{k}$, $\mathbf{g}$ and $\mathbf{k}$ later.

$$\lambda=\begin{pmatrix}\lambda_{1,1} & \cdots & \lambda_{1,l}\\ \vdots & \ddots & \\ \lambda_{l,1} & \cdots & \lambda_{l,l}\end{pmatrix}, \qquad \mathbf{g}=[g_1(x,y),\cdots,g_l(x,y)]^T, \tag{1.4}$$

Received: 24 January 2024 ; Accepted: 13 January 2025.
* Corresponding author. Email: maleknejad@iust.ac.ir .

$$\mathbf{k}(x,y,s,t) = [k_{ij}(x,y,s,t)], \quad i,j = 1,\cdots,l, \qquad \lambda\mathbf{u} = \mathbf{g} + \int_0^y \int_0^x \mathbf{ku}\,dsdt. \tag{1.5}$$

In [?], the authors obtained solutions of a two-dimensional system by Legendre wavelets. Fractional differential equations are solved by BP operational matrix in [?]. Maleknejad et al. used operational matrices based on BPFs to find the numerical solutions of stochastic Volterra integral equations [?]. Ezzati et al. applied 2D-BPfs to solve two-dimensional integro-differential equations of fractional order [?]. The system of Volterra equations has been solved by Sinc approximation in [?]. In [?] Sheng et al. and in [?] Conte and Paternoster introduced a multistep method for solving the Volterra equation. A fractional-order operational matrix method based on Euler wavelets in [?] and an operational matrix scheme based on two-dimensional wavelets in [?] for solving integro-differential equations are proposed. In [?] authors, making use of Gauss quadrature with Chebyshev polynomials for V-FIEs. Spectral Legendre-Chebyshev polynomials are used for MV-FIEs in [?] and, in [1] the chebyshev collocation method is applied for solving V-FIEs. The block-pulse method is affordable because it converts algebraic systems to a matrix system, especially sparse matrices, and reduces the amount of computation. So we studied the methods mentioned as the linear multistep method and the Rung-Kutta method and related papers. After discussing, we found that suggested methods also work like the present method in terms of convergence. We aptly elucidate the following reasons and references to confirm the claim. According to the convergence theorems in the interesting works [? ?] about the multistep collocation method and using the linear multistep method (i.e., with M=1, $m < M+1$) in both works, one can deduce that the convergence rate of the block-pulse method and multistep method are the same and equal to 1. (For comprehensive study, note to Theorem 4.2 in [?] and Theorem 4.6 in [?]). We accept that for the sake of good comparison with $hp-version$ methods (multistep methods with higher M), hybrid block-pulse and polynomials can be competitive and comparative with them as well. Suggested methods in [? ? ?] also work like the present method in terms of the convergence rate. To the best of our knowledge, different ODE and PDE problems convert to a smoother space by taking integration from them and this expression is not regular for the vise versa. It means that converting IE problems to ODE/PDE under some strong assumptions like differentiability for the unknown solution and known functions is a big hypothesis and not regular in the approximation theory. By and large, the presented method has the same convergence rate as the linear multistep method, and we ensure that the main feature of block-pulse functions as a simple tool to approximate some system problem is conducive. Simplicity of performance, less complexity and capability to improve performance, hybrid functions such as hybrid block-pulse and Legendre make them more attractive for future research.

The paper is ordered as follows: In section 2, BPfs are introduced. Section 3 centers on the product of operational matrices. A new method is proposed in section 4. Error analysis is stated in section 5. Some numerical examples in section 6 confirm the efficiency of the method and Section 7 contains the conclusion.

2. One and Two-dimensional BPfs

Consider the one-dimensional BPfs as:

$$\psi_i(x) = \begin{cases} 1, & (i-1)h \leq x \leq \ ih, \\ 0, & otherwise, \end{cases} \tag{2.1}$$

where $h = \frac{T}{m}$ and $x \in [0,T]$. The 2D-BPfs $\psi_{i,j}(x,y)(i = 1,2,...,m : j = 1,2,...,n)$ are in region of $x \in [0,T_1]$, $y \in [0,T_2]$, $h_1 = \frac{T_1}{m}$, $h_2 = \frac{T_2}{m}$, and defined as the following:

$$\psi_{i,j}(x,y) = \begin{cases} 1, & (i-1)h_1 \leq x \leq \ ih_1, (j-1)h_2 \leq y \leq \ jh_2, \\ 0, & otherwise. \end{cases} \tag{2.2}$$

Properties of 2D-BPfs:
1. Disjointness [?]

$$\psi_{i,j}(x,y)\psi_{k,l}(x,y)=\begin{cases}\psi_{i,j}(x,y), & i=k, j=l,\\ 0, & otherwise.\end{cases} \tag{2.3}$$

2. Orthogonality: For $i,k=1,2,...,m$, $j,l=1,2,...,n$,

$$\int_0^{T_1}\int_0^{T_2}\psi_{i,j}(x,y)\psi_{k,l}(x,y)dxdy=\begin{cases}h_1h_2, & i=k, j=l,\\ 0, & otherwise,\end{cases} \tag{2.4}$$

where $x\in[0,T_1]$, $y\in[0,T_2]$ and T_1, T_2 are mentioned before.
3. Completness: For every $u\in\mathbb{L}^2([0,T_1)\times[0,T_2))$ when i,j approach to the identity, Parseval's identity occurs:

$$\int_0^{T_1}\int_0^{T_2} u^2(x,y)dxdy=\sum_{i=1}^{\infty}\sum_{j=1}^{\infty}u_{i,j}^2\|\psi_{i,j}(x,y)\|^2, \tag{2.5}$$

where

$$u_{i,j}=\frac{1}{h_1h_2}\int_0^{T_1}\int_0^{T_2} u(x,y)\psi_{i,j}(x,y)dxdy. \tag{2.6}$$

3. Product of matrices

2D-BPfs can be defined as:

$$\Psi(x,y)=[\psi_{1,1}(x,y),...,\psi_{1,n}(x,y),...,\psi_{m,1}(x,y),...,\psi_{m,n}(x,y)]^T. \tag{3.1}$$

Clearly, we have:

$$\Psi(x,y).\Psi^T(x,y)=\begin{bmatrix}\psi_{1,1}(x,y) & 0 & \cdots & 0\\ 0 & \psi_{1,2}(x,y) & \cdots & 0\\ \vdots & \cdots & \ddots & \vdots\\ 0 & \cdots & 0 & \psi_{m,n}(x,y)\end{bmatrix}=\alpha, \tag{3.2}$$

$$\Psi^T(x,y).\Psi(x,y)=1, \tag{3.3}$$

$$\Psi(x,y).\Psi^T(x,y).A=\tilde{A}.\Psi(x,y), \tag{3.4}$$

$$\Psi^T(x,y).B.\Psi(x,y)=\hat{B}^T.\Psi(x,y). \tag{3.5}$$

Here, A is an mn-vector, $\tilde{A}=diag(A)$. Also, B is an $(mn)\times(mn)$-matrix and $\hat{B}$ is a mn-vector with entries equal to the diagonal entries of matrix B. Two-dimensional integration of the vector $\Psi(x,y)$ can be obtained as

$$\int_0^x\int_0^y\Psi(s,t)dsdt=E\Psi(x,y), \tag{3.6}$$

and $E = E_1 \otimes E_1$ [**?**], where $\otimes$ is a Kronecker product and E_1 is the operational matrix of BPfs defined over $[0, 1]$ with $h = \frac{1}{m}$ as follows:

$$E_1 = \frac{h}{2}\begin{bmatrix} 1 & 2 & 2 & \cdots & 2 \\ 0 & 1 & 2 & \cdots & 2 \\ 0 & 0 & 1 & \cdots & 2 \\ \cdots & \cdots & \cdots & \cdots & \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}. \tag{3.7}$$

So, by relation (3.1) we have:

$$\int_0^T \int_0^T \Psi(x,y)\Psi^T(x,y)dxdy = \int_0^T \int_0^T \alpha dxdy = h^2 I = \begin{bmatrix} h^2 & 0 & 0 & \cdots & 0 \\ 0 & h^2 & 0 & \cdots & 0 \\ \vdots & \cdots & \vdots & \cdots & \\ 0 & 0 & 0 & \cdots & h^2 \end{bmatrix}. \tag{3.8}$$

4. Explanation of the method

We apply 2D-BPfs to solve the Eq. (1.1). Suppose that

$$u_j(x,y) \simeq C_j\Psi(x,y), \qquad \begin{bmatrix} u_1(x,y) \\ u_2(x,y) \\ \vdots \\ u_l(x,y) \end{bmatrix} = \begin{bmatrix} C_1\Psi(x,y) \\ C_2\Psi(x,y) \\ \vdots \\ C_l\Psi(x,y) \end{bmatrix}, \tag{4.1}$$

where $\Psi(x,y)$ is defined in the previous section and $C_j, j = 1, 2, ..., l$ are unknowns.

$$u_j(x,y) \simeq \sum_{i=1}^{m}\sum_{j=1}^{l} c_{ij}\psi_{i,j}(x,y), \tag{4.2}$$

where

$$c_{ij} = \frac{1}{h_1 h_2}\int_{(i-1)h_1}^{ih_1}\int_{(j-1)h_2}^{jh_2} u_j(x,y)dxdy. \tag{4.3}$$

Then Eq. (1.1) converts to:

$$\sum_{j=1}^{l}\lambda_{ij}\sum_{i=1}^{m}\sum_{j=1}^{l} c_{ij}\psi_{i,j}(x,y) = g_i(x,y) + \sum_{j=1}^{l}\int_0^y\int_0^x k_{ij}(x,y,s,t)\sum_{i=1}^{m}\sum_{j=1}^{l} c_{ij}\psi_{i,j}(s,t)dsdt.$$

Suppose $k_{im}(x,y,s,t)$ is a function of four variables in $[0,T_1]\times[0,T_2]\times[0,T_3]\times[0,T_4]$. It can be expanded with respect to 2D-BPfs as:

$$k_{im}(x,y,s,t) \simeq \Psi^T(x,y)K_{im}\Psi(s,t), \tag{4.4}$$

where $\Psi(x,y), \Psi(s,t)$ are 2D-BPfs vectors with mn, kl dimensions, respectively, and K is the $(mn)\times(kl)$ 2D-BPfs functions matrix.

$$k_{im} = [k_{ijmn}], j, n = 1 : l,$$

$$k_{ijmn} = \frac{<\psi_{i,j}(x,y), < k_{im}, \psi_{mn}(s,t) >>}{\|\psi_{i,j}(x,y)\|^2\|\psi_{m,n}(x,y)\|^2}, \tag{4.5}$$

$$\mathbf{g}(x,y) \simeq {G_j}^T\Psi(x,y) = \Psi^T(x,y)G_j. \tag{4.6}$$

By using (4.4) and (4.6), we have:

$$\begin{aligned}\int_0^y\int_0^x k_{ij}(x,y,s,t)u_j(s,t)dsdt &= \int_0^y\int_0^x \Psi^T(x,y)K_{ij}\Psi(s,t)\Psi^T(s,t)C_j dsdt\\ &= \int_0^y\int_0^x \Psi^T(x,y)k_{ij}\tilde{C}_j\Psi(s,t)dsdt\\ &= \Psi^T(x,y)K_{ij}\tilde{C}_j\int_0^y\int_0^x \Psi(s,t)dsdt\\ &= \Psi^T(x,y)K_{ij}\tilde{C}_jE\Psi(x,y)\\ &= \Psi^T(x,y)B_{ij}\Psi(x,y)\\ &= \hat{B}_{ij}^T\Psi(x,y).\end{aligned}$$

In above equation

$$K_{ij}\tilde{C}_jE = B_{ij}. \tag{4.7}$$

Therefore

$$\int_0^y\int_0^x k_{ij}(x,y,s,t)u_j(s,t)dsdt = \hat{B}_{ij}^T\Psi(x,y). \tag{4.8}$$

So, Eq. (1.1) becomes:

$$\begin{bmatrix}\lambda_{11} & \cdots & \lambda_{1l}\\ \vdots & \ddots & \vdots\\ \lambda_{l1} & \cdots & \lambda_{ll}\end{bmatrix}\begin{bmatrix}u_1(x,y)\\ \vdots\\ u_l(x,y)\end{bmatrix} = \begin{bmatrix}g_1(x,y)\\ \vdots\\ g_l(x,y)\end{bmatrix} + \int_0^y\int_0^x \begin{bmatrix}k_{11}(x,y,s,t) & \cdots & k_{1l}(x,y,s,t)\\ \vdots & \ddots & \vdots\\ k_{l1}(x,y,s,t) & \cdots & k_{ll}(x,y,s,t)\end{bmatrix}\begin{bmatrix}u_1(s,t)\\ \vdots\\ u_l(s,t)\end{bmatrix}dsdt.$$

Then by relations (4.7) and (4.8) we have:

$$\lambda\Psi^T(x,y)C = \Psi^T(x,y)G + \Psi^T(x,y)\hat{B}. \tag{4.9}$$

The new system becomes $\lambda C = G + \hat{B}$. Now, we construct a system and solve it by Newton's method.

5. Convergence and Error analysis

Now we present error analysis of the method on $\rho = [0,1]^4$. For convenience, we put $m_1 = m_2 = m$, and consequently $h_1 = h_2 = \frac{1}{m}$. Also, we consider for simplicity the $2-$norm for matrixes.

$$\mathbf{u}(x,y) = [u_j(x,y)], \quad j = 1,2,...,l, \tag{5.1}$$

$$\mathbf{g}(x,y) = [g_j(x,y)], \quad j = 1,2,...,l, \tag{5.2}$$

and

$$\mathbf{k}(x,y) = [k_{i,j}(x,y,s,t)], \quad i,j = 1,2,...,l, \tag{5.3}$$

as

$$\|f\|_2 = \left[\int_0^1\int_0^1 |f(x,y)|^2dxdy\right]^{\frac{1}{2}},$$

$$\|k\|_2 = \left[\int_0^1\int_0^1\int_0^1\int_0^1 |k(x,y,s,t)|^2dxdydsdt\right]^{\frac{1}{2}},$$

so

$$\|\mathbf{f}\|_2 = \left(\sum_{j=1}^{l} \|f_j\|_2^2 \right)^{\frac{1}{2}} \tag{5.4}$$

and

$$\|\mathbf{k}\|_2 = \left(\sum_{i=1}^{l} \sum_{j=1}^{l} \|K_{i,j}\|_2^2 \right)^{\frac{1}{2}}, \tag{5.5}$$

for every $(x, y) \in \tau = [0, 1]^2$ and $(x, y, s, t) \in \rho$.

Theorem 5.1. *[**?**] Suppose $f(x, y)$, is a differentiable function on τ with*

$$\|f'\|_2 \leq \alpha.$$

Let

$$\hat{f}_m(x, y) = \sum_{i=1}^{m} \sum_{j=1}^{m} f_{ij} \psi_i(x) \psi_j(y),$$

be the 2D-BPfs expansion of $f(x, y)$ and $e(x, y) = f(x, y) - \hat{f}_m(x, y)$, then for every $(x, y) \in \tau$ we have

$$\|e\|_2^2 \leq \frac{2}{m^2} \times \alpha^2,$$

hence

$$\|e\|_2 = O(\frac{1}{m}).$$

Theorem 5.2. *[**?**] Suppose that $k(x, y, s, t)$, is defined on ρ with*

$$\|k'\|_2 \leq \beta,$$

and

$$\hat{k}_m(x, y, s, t) = \sum_{i=1}^{m} \sum_{j=1}^{m} \sum_{p=1}^{m} \sum_{q=1}^{m} k_{ijpq} \psi_i(x) \psi_j(y) \psi_p(s) \psi_q(t),$$

is four-dimensional BPfs expansion of $k(x, y, s, t)$. If

$$e(x, y, s, t) = k(x, y, s, t) - \hat{k}_m(x, y, s, t),$$

then for every $(x, y, s, t) \in \rho$ we have

$$\|e\|_2^2 \leq \frac{4}{m^2} \times \beta^2,$$

hence

$$\|e\|_2 = O(\frac{1}{m}).$$

Theorem 5.3. *Let $\boldsymbol{f}(x, y)$ be as defined in (5.4) and $\boldsymbol{f}_m(x, y)$ be the BPfs matrix of $\boldsymbol{f}(x, y)$. Then for $(x, y) \in \tau$ we have*

$$\|\boldsymbol{f} - \boldsymbol{f}_m\|_2 \leq \frac{C}{m},$$

where

$$C = \left(\sum_{j=1}^{l} 2\alpha_j^2 \right)^{\frac{1}{2}}.$$

Proof. According to the defined norm in (5.4), we write

$$\|\mathbf{f} - \mathbf{f}_m\|_2 = \left(\sum_{j=1}^{l} \|f_j - f_m^j\|_2^2 \right)^{\frac{1}{2}},$$

from Theorem 5.1, we conclude

$$\|\mathbf{f} - \mathbf{f}_m\|_2 \leq \left(\sum_{j=1}^{l} \frac{2}{m^2} \times \alpha_j^2 \right)^{\frac{1}{2}} = \frac{1}{m} \left(\sum_{j=1}^{l} 2\alpha_j^2 \right)^{\frac{1}{2}} = \frac{C}{m},$$

where

$$C = \left(\sum_{j=1}^{l} 2\alpha_j^2 \right)^{\frac{1}{2}}.$$

□

Theorem 5.4. *Let $\boldsymbol{k}(x, y, s, t)$ be as defined in (5.3) and $\boldsymbol{k}_m(x, y, s, t)$ be the BPfs matrix of $\boldsymbol{k}(x, y, s, t)$. Then for $(x, y, s, t) \in \rho$ we have*

$$\|\boldsymbol{k} - \boldsymbol{k}_m\|_2 \leq \frac{C'}{m},$$

so

$$C' = \left(\sum_{i=1}^{l} \sum_{j=1}^{l} 4\beta_{i,j}^2 \right)^{\frac{1}{2}},$$

where β is defined in Theorem 5.2.

Proof. The proof is similar to the proof of Theorem 5.3 by using Theorem 5.2.

□

Theorem 5.5. *Suppose that $\boldsymbol{u}(x, y)$, is the exact and $\boldsymbol{u}_m(x, y)$ is the BPfs approximate solution of Eq. (1.1), respectively. Also, assume that*

(a) $\|\boldsymbol{u}\|_2 \leq \sigma$, $(x, y) \in \tau$,
(b) $\|\boldsymbol{k}\|_2 \leq \sigma'$, $(x, y, s, t) \in \rho$,
(c) $(\sigma' + \frac{C'}{m}) < \|\lambda\|_2$,

then

$$\|\boldsymbol{u} - \boldsymbol{u}_m\|_2 = O\left(\frac{1}{m}\right).$$

Proof. By considering (a), we have

$$\lambda\left(\mathbf{u}(x,y) - \mathbf{u}_m(x,y)\right) = \mathbf{g}(x,y) - \mathbf{g}_m(x,y) + \int_0^y \int_0^x \left(\mathbf{k}(x,y,s,t)\mathbf{u}(s,t) - \mathbf{k}_m(x,y,s,t)\mathbf{u}_m(s,t)\right) dsdt,$$

then by mean value theorem for the 2D-integrals, for every $(x,y) \in \tau$ and $(x,y,s,t) \in \rho$, we have

$$\|\lambda\|_2\|\mathbf{u} - \mathbf{u}_m\|_2 \leq \|\mathbf{g} - \mathbf{g}_m\|_2 + xy\|\mathbf{k}\mathbf{u} - \mathbf{k}_m\mathbf{u}_m\|_2. \tag{5.6}$$

By using hypothesises (a) and (b) and Theorem 5.4, we get

$$\|\mathbf{k}\mathbf{u} - \mathbf{k}_m\mathbf{u}_m\|_2 \leq \|\mathbf{k}\|_2\|\mathbf{u} - \mathbf{u}_m\|_2 + \|\mathbf{k} - \mathbf{k}_m\|_2\left(\|\mathbf{u} - \mathbf{u}_m\|_2 + \|\mathbf{u}\|_2\right)$$

$$\leq \sigma'\|\mathbf{u} - \mathbf{u}_m\|_2 + \frac{C'}{m}\left(\|\mathbf{u} - \mathbf{u}_m\|_2 + \sigma\right). \tag{5.7}$$

By substituting (5.6) in (5.7) and using Theorem 5.3, we have

$$\|\lambda\|_2\|\mathbf{u} - \mathbf{u}_m\|_2 \leq \frac{C}{m} + xy\left[(\sigma' + \frac{C'}{m})\|\mathbf{u} - \mathbf{u}_m\|_2 + \frac{C'}{m}\sigma\right],$$

by taking *sup*, we can get

$$\|\lambda\|_2\|\mathbf{u} - \mathbf{u}_m\|_2 \leq \frac{C}{m} + 1 \times 1\left[(\sigma' + \frac{C'}{m}) \times sup_{(x,y)\in\tau}\|\mathbf{u} - \mathbf{u}_m\|_2 + \frac{C'}{m}\sigma\right],$$

where by considering hypothesis (c) we have

$$\|\mathbf{u} - \mathbf{u}_m\|_2 \leq \frac{\frac{C}{m} + \frac{C'}{m}\sigma}{\|\lambda\|_2 - (\sigma' + \frac{C'}{m})},$$

hence

$$\|\mathbf{u} - \mathbf{u}_m\|_2 = O\left(\frac{1}{m}\right).$$

Now the accuracy of the solution is being discussed. When the approximated solution of Eq. (1.1) is substituted in it, the error is obtained. That is, for $0 \leq a \leq x, y \leq b$, $(x,y) = (x_p, y_p)$, $p = 0, 1, 2, \ldots$:

$$E_j(x_p, y_p) = \|\sum_{j=1}^{l} \lambda_{ij}u_j(x_p, y_p) - g_i(x_p, y_p) - \sum_{j=1}^{l}\int_0^{y_p}\int_0^{x_p} k_{ij}(x_p, y_p, s, t)u_j(s,t)dsdt\| \simeq 0, \tag{5.8}$$

$i, j = 1, 2, \ldots, l.$, $E_j(x_p, y_p) \leq 10^{-m_p}$ (m_p is a positive integer). If $\max 10^{-m_p} = 10^{-m}$ (m is a positive integer), then the error can be estimated by

$$E_N(x,y) = \Lambda(x,y)U_N(x,y) - G(x,y) - KU. \tag{5.9}$$

If N is large enough, then the error decreases. □

TABLE 1. Example 6.1, $m = 10$ and $T_1 = T_2 = 1$.

(x_i, y_i)	Exact $(u_1(x,y), u_2(x,y))$	Approximation $(u_1(x,y), u_2(x,y))$	error $(u_1(x,y), u_2(x,y))$	$l^2 error$
$(\frac{1}{11}, \frac{1}{11})$	$(2.48902 \times 10^{-2}, 6.20921 \times 10^{-6})$	$(1.31464 \times 10^{-2}, 8.33337 \times 10^{-7})$	$(1.17438 \times 10^{-2}, 5.37588 \times 10^{-6})$	(1.17438×10^{-2})
$(\frac{2}{11}, \frac{2}{11})$	$(5.4518 \times 10^{-2}, 1.98695 \times 10^{-4})$	$(4.35877 \times 10^{-2}, 8.75053 \times 10^{-5})$	$(1.09303 \times 10^{-2}, 1.1119 \times 10^{-4})$	$(1.101023 \times 10^{-2})$
$(\frac{3}{11}, \frac{3}{11})$	$8.95597 \times 10^{-2}, 1.50884 \times 10^{-3})$	$(8.02845 \times 10^{-2}, 1.02948 \times 10^{-3})$	$(9.27513 \times 10^{-3}, 4.79361 \times 10^{-4})$	(9.301×10^{-4})
$(\frac{4}{11}, \frac{4}{11})$	$(1.30777 \times 10^{-1}, 6.35823 \times 10^{-3})$	$(1.2419 \times 10^{-1}, 5.40078 \times 10^{-3})$	$(6.58714 \times 10^{-3}, 9.57454 \times 10^{-4})$	(6.58714×10^{-6})
$(\frac{5}{11}, \frac{5}{11})$	$(1.79029 \times 10^{-1}, 1.94038 \times 10^{-2})$	$(1.76325 \times 10^{-1}, 1.87978 \times 10^{-2})$	$(2.70376 \times 10^{-3}, 6.05944 \times 10^{-4})$	$(1.047728 \times 10^{-3})$
$(\frac{6}{11}, \frac{6}{11})$	$(2.35281 \times 10^{-1}, 4.82828 \times 10^{-2})$	$(2.37745 \times 10^{-1}, 5.11039 \times 10^{-2})$	$(2.46374 \times 10^{-3}, 2.8211 \times 10^{-3})$	(1.26×10^{-3})
$(\frac{7}{11}, \frac{7}{11})$	$3.00618 \times 10^{-1}, 1.04358 \times 10^{-1})$	$(3.09552 \times 10^{-1}, 1.17859 \times 10^{-1})$	$(8.93441 \times 10^{-3}, 1.35007 \times 10^{-2})$	(3.12348×10^{-3})
$(\frac{8}{11}, \frac{8}{11})$	$(3.7626 \times 10^{-1}, 2.03463 \times 10^{-1})$	$(3.93078 \times 10^{-1}, 2.41836 \times 10^{-1})$	$(1.68178 \times 10^{-2}, 3.83725 \times 10^{-2})$	$(1.327296 \times 10^{-2})$
$(\frac{9}{11}, \frac{9}{11})$	$(4.63577 \times 10^{-1}, 3.66648 \times 10^{-1})$	$(4.90387 \times 10^{-1}, 4.54979 \times 10^{-})$	$(2.68101 \times 10^{-2}, 8.83312 \times 10^{-2})$	(9.25×10^{-2})
$(\frac{10}{11}, \frac{10}{11})$	$(5.64106 \times 10^{-1}, 6.20921 \times 10^{-1})$	$(6.05424 \times 10^{-1}, 8.00921 \times 10^{-1})$	$(4.13188 \times 10^{-2}, 1.8 \times 10^{-1})$	(1.803×10^{-2})

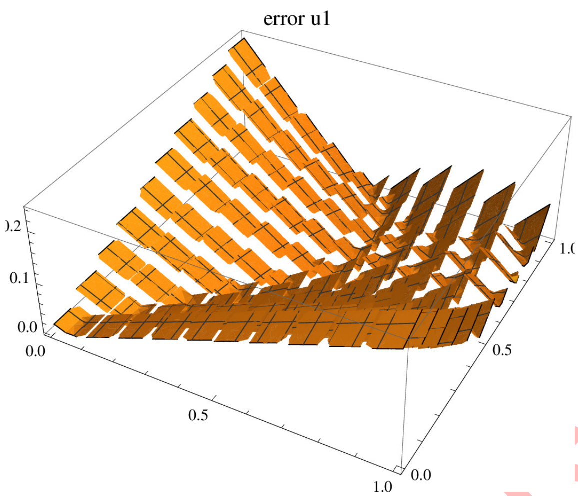

FIGURE 1. Error Example 6.1 u_1 for $m = 10$ and $T_1 = T_2 = 1$.

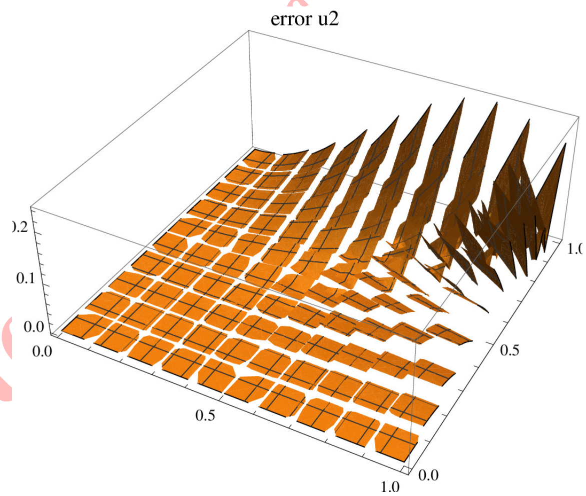

FIGURE 2. Error Example 6.1 u_2 for $m = 10$ and $T_1 = T_2 = 1$.

6. EXAMPLES

Example 6.1. Consider

$$\begin{cases} u_1(x,y) - u_2(x,y) = g_1(x,y) + \int_0^y \int_0^x 2xu_1(s,t) + e^{-t}u_2(s,t)dtds \\ -u_2(x,y) = g_2(x,y) + \int_0^y \int_0^x s^2t^3u_1(s,t) + tyu_2(s,t)dtds, \end{cases} \tag{6.1}$$

$$K(x,y,s,t) = \begin{pmatrix} 2x & e^{-t} \\ s^2t^3 & ty \end{pmatrix}, \qquad \lambda = \begin{pmatrix} 1 & -1 \\ 0 & -1 \end{pmatrix},$$

$$g_1(x,y) = \frac{1}{12}x\Big(4x^2e^{-y}\big(y^3 + 3y^2 + 6y + 6\big) - 3\big(x^2 - 1\big)e^y - 3x\big(7x + 4y^3\big)\Big),$$

$$g_2(x,y) = -\frac{1}{16}x^4\Big(e^y(y((y-3)y+6)-6)+6\Big) - \frac{1}{15}x^3y^6 - x^2y^3,$$

with the exact solutions $u_1(x,y) = \frac{1}{4}xe^y$ and $u_2(x,y) = x^2y^3$. The approximate solutions obtained by BPfs and the exact solutions for different values of m and T_1, T_2 and l^2 error are in Table I and Figures 1, 2.

TABLE 2. Example 6.2, $m = 10$ and $T_1 = T_2 = 1$.

(x_i, y_i)	$u_{exact}(x,y)$	$u_{approximate}(x,y)$	Error $u(x,y)$	$l^2 error$
$(\frac{1}{11}, \frac{1}{11})$	1.65289×10^{-2}	6.66673×10^{-3}	9.8622×10^{-3}	1.54×10^{-2}
$(\frac{2}{11}, \frac{2}{11})$	6.61157×10^{-2}	4.66702×10^{-2}	1.94455×10^{-2}	2.48139×10^{-2}
$(\frac{3}{11}, \frac{3}{11})$	1.4876×10^{-1}	1.26693×10^{-1}	2.2067×10^{-2}	3.539×10^{-2}
$(\frac{4}{11}, \frac{4}{11})$	2.64463×10^{-1}	2.46768×10^{-1}	7.65×10^{-3}	7.64×10^{-3}
$(\frac{5}{11}, \frac{5}{11})$	4.13223×10^{-1}	4.06944×10^{-1}	6.27943×10^{-3}	6.28×10^{-3}
$(\frac{6}{11}, \frac{6}{11})$	5.95041×10^{-1}	6.07288×10^{-1}	1.2247×10^{-2}	1.237×10^{-2}
$(\frac{7}{11}, \frac{7}{11})$	8.09917×10^{-1}	8.47894×10^{-1}	3.79766×10^{-2}	3.7987×10^{-2}
$(\frac{8}{11}, \frac{8}{11})$	1.05785	1.12889	7.10343×10^{-2}	7.004×10^{-2}
$(\frac{9}{11}, \frac{9}{11})$	1.33884	1.45044	1.11595×10^{-1}	1.1200×10^{-1}
$(\frac{10}{11}, \frac{10}{11})$	1.65289	1.81281	1.59915×10^{-1}	1.6021×10^{-1}

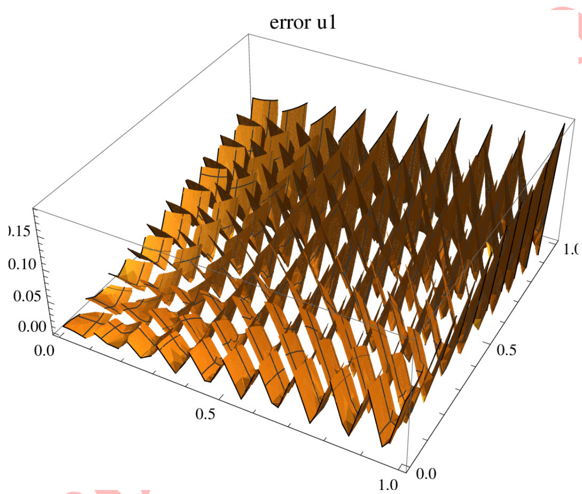

FIGURE 3. Error Example 6.2 for $m = 10$ and $T_1 = T_2 = 1$.

Example 6.2. Consider

$$u(x,y) = g(x,y) + \int_0^y \int_0^x (s^2 + t^2) u(s,t) ds dt, \tag{6.2}$$

$$K(x,y,s,t) = s^2 + t^2, \qquad \lambda = 1, \qquad g(x,y) = x^2 - \frac{1}{45} xy \left(9x^4 + 10x^2y^2 + 9y^4\right) + y^2, \tag{6.3}$$

and $u_{exact}(x,y) = x^2 + y^2$. Table 2 and Figure 3 show the results.

Example 6.3. We consider a problem that was solved numerically in [?]. In this case, the kernel is not separable. We have:

$$\lambda(x,y) = \begin{pmatrix} 1 & x+y \\ x+y & (x+y)^2 \end{pmatrix}, \tag{6.4}$$

TABLE 3. Example 6.3, $m = 10$ and $T_1 = T_2 = 1$.

(x_i, y_i)	Exact $(u_1(x,y), u_2(x,y))$	Approximation $(u_1(x,y), u_2(x,y))$	error $(u_1(x,y), u_2(x,y))$	$l^2 error$
$(\frac{1}{11}, \frac{1}{11})$	$(2.48902 \times 10^{-2}, 6.20921 \times 10^{-6})$	$(1.31464 \times 10^{-2}, 8.33337 \times 10^{-7})$	$(1.17438 \times 10^{-2}, 5.37588 \times 10^{-6})$	(1.17438×10^{-2})
$(\frac{2}{11}, \frac{2}{11})$	$(5.4518 \times 10^{-2}, 1.98695 \times 10^{-4})$	$(4.35877 \times 10^{-2}, 8.75053 \times 10^{-5})$	$(1.09303 \times 10^{-2}, 1.1119 \times 10^{-4})$	(1.09303×10^{-2})
$(\frac{3}{11}, \frac{3}{11})$	$(8.95597 \times 10^{-2}, 1.50884 \times 10^{-3})$	$(8.02845 \times 10^{-2}, 1.02948 \times 10^{-3})$	$(9.27513 \times 10^{-3}, 4.79361 \times 10^{-4})$	$(9.287509 \times 10^{-3})$
$(\frac{4}{11}, \frac{4}{11})$	$(1.30777 \times 10^{-1}, 6.35823 \times 10^{-3})$	$(1.2419 \times 10^{-1}, 5.40078 \times 10^{-3})$	$(6.58714 \times 10^{-3}, 9.57454 \times 10^{-4})$	(6.65636×10^{-3})
$(\frac{5}{11}, \frac{5}{11})$	$(1.79029 \times 10^{-1}, 1.94038 \times 10^{-2})$	$(1.76325 \times 10^{-1}, 1.87978 \times 10^{-2})$	$(2.70376 \times 10^{-3}, 6.05944 \times 10^{-4})$	(2.85150×10^{-3})
$(\frac{6}{11}, \frac{6}{11})$	$(2.35281 \times 10^{-1}, 4.82828 \times 10^{-2})$	$(2.37745 \times 10^{-1}, 5.11039 \times 10^{-2})$	$(2.46374 \times 10^{-3}, 2.8211 \times 10^{-3})$	(3.74547×10^{-3})
$(\frac{7}{11}, \frac{7}{11})$	$(3.00618 \times 10^{-1}, 1.04358 \times 10^{-1})$	$(3.09552 \times 10^{-1}, 1.17859 \times 10^{-1})$	$(8.93441 \times 10^{-3}, 1.35007 \times 10^{-2})$	(1.61892×10^{-2})
$(\frac{8}{11}, \frac{8}{11})$	$(3.7626 \times 10^{-1}, 2.03463 \times 10^{-1})$	$(3.93078 \times 10^{-1}, 2.41836 \times 10^{-1})$	$(1.68178 \times 10^{-2}, 3.83725 \times 10^{-2})$	(4.18961×10^{-2})
$(\frac{9}{11}, \frac{9}{11})$	$(4.63577 \times 10^{-1}, 3.66648 \times 10^{-1})$	$(4.90387 \times 10^{-1}, 4.54979 \times 10^{-1})$	$(2.68101 \times 10^{-2}, 8.83312 \times 10^{-2})$	(9.23102×10^{-2})
$(\frac{10}{11}, \frac{10}{11})$	$(5.64106 \times 10^{-1}, 6.20921 \times 10^{-1})$	$(6.05424 \times 10^{-1}, 8.00921 \times 10^{-1})$	$(4.13188 \times 10^{-2}, 1.8 \times 10^{-1})$	(1.84681×10^{-1})

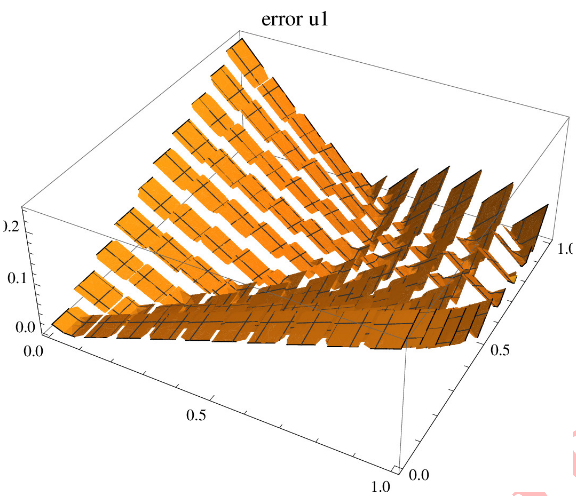

FIGURE 4. Error $u_1(x)$ Example 6.3 for $m = 10$ and $T_1 = T_2 = 1$.

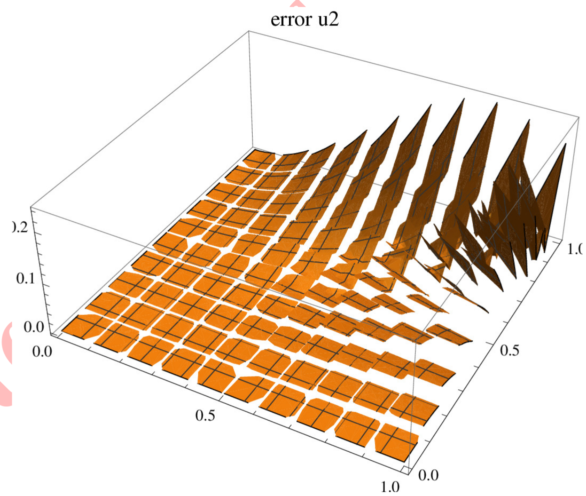

FIGURE 5. Error $u_2(x)$ Example 6.3 for $m = 10$ and $T_1 = T_2 = 1$.

$$G(t,x) = \begin{pmatrix} \frac{t^4 x}{12} + \frac{t^3 x^2}{2} - \frac{t^2 x^2}{4} + \frac{t^2 x}{2} + t^2 - \frac{5tx^3}{6} + \frac{3tx^2}{2} + e^{-t-x}(t+x) - x + 1 \\ \frac{t^3 x}{3} + \left(t^2 - x + 1\right)(t+x) - \frac{tx^2}{2} + 5txe^{t-x} + tx + e^{-t-x}(t+x)^2 \end{pmatrix}, \tag{6.5}$$

$$K(t,x,\tau,s) = -\begin{pmatrix} s + t - \tau + x & 0 \\ 1 & 5\exp(s + t + \tau - x) \end{pmatrix}, \tag{6.6}$$

where u_{exact} is

$$u(t,x) = \begin{pmatrix} t^2 - x + 1 \\ e^{-t-x} \end{pmatrix}.$$

Here $m = 10$ and $T_1 = T_2 = 1$ and the numerical results and l^2 error are reported in Table III and Figures 4,5.

As we said, this example has been solved in [**?**]. The error norms are given by $\|\epsilon_{h,q}\|_\infty = max\|u_{ij} - u(t_i, x_j)\|_\infty$,$i = 1, 2, ..., N,\ j = 1, 2, ..., M$. Also, the estimates of convergence orders are $p = \log_2 \frac{\|\epsilon_{h,q}\|_\infty}{\|\epsilon_{\frac{h}{2},\frac{q}{2}}\|_\infty}$. Besides $max\|u_{ij} - u(t_i, x_j)\| = O(h + q)$. ϵ is in the interval of $(0.00536162, 0.626502)$, here we have the same error with fewer meshes.

Authors point out that a low convergence rate is compensated by the simplicity of computations. A comparison of methods is between our method and method in [?]. In [?] l^2 errors are in the interval $(7.17476 \times 10^{-2}, 9.49 \times 10^{-1})$ but, in our method, l^2 errors are in the interval $(2.85150 \times 10^{-3}, 1.84681 \times 10^{-1})$. This means that our method is more precise than the method in [?]. Also, in [?] the error norms are in $[10^{-1}, 10^{-3}]$ in 3 examples.

7. Conclusion

We have successfully applied BPfs to obtain the solution of a system of 2D-VIE. The new way is suitable for two kinds of kernels, such as separable or not. Other types of systems of integral equations can be solved by this method in future work. The method has the same convergence rate as the linear multistep method. Simplicity of performance, less complexity and capability improve hybrid functions such as hybrid block-pulse and Legendre and make them more attractive for future research.

References

[1] W. M. Abd-Elhameed and Y. H. Youssri, *Numerical solutions for Volterra-Fredholm-Hammerstein integral equations via second kind Chebyshev quadrature collocation algorithm*, Adv. math. sci. appl, (2014).

[2] K. Atkinson and W. Han, *Theoretical numerical analysis*, Third ed., Applied Mathematics. Springer, Dordrecht, *39* (2005).

[3] S. Behera and S. Saha Ray, *Euler wavelet method for solving fractional-order linear Volterra-Fredholm integro-differential equations with weakly singular kernels*, J CAPM, *40*(192) (2021).

[4] S. Behera and S. Saha Ray, *An operational matrix based scheme for numerical solutions of nonlinear weakly singular partial integro-differential equations*, J. Appl. Math. Comp, *367* (2020), 124771.

[5] I. Brunner, *Collocation Methods for Volterra Integral and Related Functional Differential Equations*, Cambridge University Press, Cambridge, (2004).

[6] M. V. Bulatov and P. M. Lima, *Two-dimensional integral-algebraic systems: Analysis and computational methods*, J. Comput. Appl. Math. *236* (2011), 132–140.

[7] D. Conte and B. Paternoster, *Multistep collocation methods for Volterra integral equations*, J. Appl. Num. Math, *59*(8) (2009), 1721-1736.

[8] G. V. Demidenko and S. V. Uspenski, *Partial Differential Equations and Systems not Solvable with Respect to the Highest-order Derivative*, Taylor and Francis, (2005).

[9] M. Fallahpour, M. Khodabin, and K. Maleknejad, *Theoretical error analysis and validation in numerical solution of two-dimensional linear stochastic Volterra-Fredholm integral equation by applying the block-pulse functions*, J. Cogent Mathematics. *4* (2017).

[10] R. M. Hafez and Y. H. Youssri,*Spectral Legendre-Chebyshev treatment of 2D linear and nonlinear mixed Volterra-Fredholm integral equation*, J. Math. Sci. Lett, (2020).

[11] A. Karimi, K. Maleknejad, and R. Ezzati,*Numerical solutions of system of two-dimensional Volterra integral equation via Legendre wavelets and convergence*, J. Appl. Num. Math, *156* (2020), 228-241.

[12] Y. Li and N. Sun, *Numerical solution of fractional differential equations using the generalized block pulse operational matrix.*, J. Comput. Math. Appl. *62* (2011), 1046-1054.

[13] K. Maleknejad, M. Khodabin, and M. Rostami, *Numerical solution of stochastic Volterra integral equations by a stichastic operational matrix based on block pulse functins.*, J. Math. Comput. Model. *55* (2012), 791-800.

[14] K. Maleknejad and A. Ostadi, *Numerical solution of system of Volterra integral equations with weakly singular kernels and its convergence analysis.*, J. Appl. Numer. Math. *115* (2017), 82-98.

[15] B. Moiseiwitsch, *Integral Equations*, Dover Publications Inc., New York, (2005).

[16] S. Najafalizadeh and R. Ezzati, *A block puls nonlinear eoperational matrix method for solving two-dimensional nonlinear integro-differential equations of fractional order*, J. Comput. Appl. Math. *326* (2017), 159-170.

[17] Y. Qian, I. Wu, and Y. Zhang, *A Note on the Unique Solutionof the Integral Equations in the framework of Fixed Point Theorem on Partially Ordered Metric Space*, Applied Mathematics and Statistics, *4* (5) (2016), 154-160.

[18] C. T. Sheng, Z. Q. Wang, and B. Y. Guo, *A Multistep Legendre–Gauss Spectral Collocation Method for Nonlinear Volterra Integral Equations*, SIAM Journal on Numerical Analysis, *52*(4) (2014), 1953-1980.

[19] Y. H. Youssri and R. M. Hafez, *Chebyshev collocation treatment of Volterra–Fredholm integral equation with error analysis*, Arabian Journal of Mathematics, (2020) Springer.