ISSN 2229-5518

# Chebyshev-Collocation Spectral Method for Class of Parabolic-Type Volterra Partial Integro-Differential Equations

G. I. El-Baghdady, M. S. El-Azab, W. S. El-Beshbeshy.

**Abstract**— The main purpose of this paper is to introduce a novel numerical method for parabolic Volterra partial integro-differential equations based on Chebyshev-collocation spectral scheme. In the present work, the parabolic Volterra integro-differential equation is converted to two coupled equivalent Volterra equations of the second kind. Then, we approximate the integration by replacing the integral function by its interpolating polynomials with Lagrange basis functions in terms of the Chebyshev polynomials instead of using Gauss quadrature approximation to obtain a linear algebraic system. Finally, some numerical examples are presented to illustrate the efficiency and accuracy of the proposed method.

**Index Terms**— Chebyshev spectral collocation method, Differentiation matrix, Lagrange basis function, Parabolic Volterra integro-differential equations (PVIDE).

---

## 1 Introduction

CONSIDER the one dimensional parabolic partial Volterra integro-differential equation

$$\partial_t u - \Delta u = \int_0^t k(x,t-s)\Delta u(x,s)\,ds + f(x,t), \qquad (1)$$

with the initial condition

$$u(x,0) = u_0(x), \quad x \in \Omega, \qquad (2)$$

and Dirichlet boundary conditions are assigned on the boundary

$$u = 0, \quad (x,t) \in \partial\Omega \times I, \qquad (3)$$

where $\Delta$ is the Laplace operator in $(x,t) \in Q \equiv \Omega \times I$, $\Omega$ is the bounded interval $[-1,\ 1]$, with boundary $\partial\Omega \in \{-1,1\}$ and $I \equiv (0,\ T)$ for a given fixed number $T > 0$. For implementation of high-order methods such as spectral methods, the known functions *f*, *k* and $u_0(x)$ are assumed to be sufficiently smooth; real valued functions.

Problems of type (1)-(3) arise in many applications; for example, it describes compression of poro-viscoelastic media [1], [2], the nonlocal reactive flows in porous media [4], [5], [6] and heat conduction through materials with memory term [2], [3].

Many authors have been considered the numerical solution of partial Volterra integro-differential equations by many methods; for example, in [7] (Amiya K. Pani, et al.) use Galerkin finite element method and ADI orthogonal spline collocation in [8], [9] and the references therein. In [10] F. Fakhar-Izadi and M. Dehghan apply Legendre spectral method to (PVIDE).

Spectral methods have a considerable attention in the last few years; see [11], [12], [13], [14], and [15]. Spectral methods are nice and powerful approach for the numerical solution for ordinary or partial differential equations to high accuracy on a simple domain and if the data defining the problem are smooth, see [16]. Also Volterra integral and ordinary Volterra integro-differential equations have a wide interest by using many methods. In [17] H. Brunner used Collocation methods for second-order Volterra integro-differential equations. Multistep collocation method is also used for Volterra integral equations in [18]. Chebyshev spectral collocation method for the solution of Volterra integral and ordinary Volterra integro-differential equations are discussed in [19]. In [20] Tang introduces Legendre-spectral method with its error analysis for ordinary Volterra integro-differential equation of the second kind. Another spectral method using Legendre spectral Galerkin method was introduced for second-kind Volterra integral equations in [21]. Most papers that mentioned before were devoted to VIEs and ordinary VIDEs, but in this article we considered the PVIDEs.

Our goal in this article is to apply a Chebyshev-collocation method for both space and time variables that are an extension of the method presented in [22], [23].

The organization of this paper is as follows. In Section 2, we present the Chebyshev-collocation spectral scheme for discretizing the introduced problem. As a result a set of algebraic linear equations are formed and a solution of the considered problem is discussed. In Section 3, we present some nu-


- G. I. El-Baghdady is currently Ph. student degree in Engineering Physics & Mathematics Dept., Faculty of Engineering, Mansoura Univ., Egypt, PH-+201005208927. E-mail: amoun1973@yahoo.com
- M. S. El-Azab is currently Prof. in Engineering Physics & Mathematics Dept., Faculty of Engineering, Mansoura University, Egypt, PH-+201227379809. E-mail: ms_elazab@hotmail.com
- W. S. El-Beshbeshy is currently Dr. in the same Dept., E-mail: welbeshbeshy@yahoo.com

merical examples to demonstrate the effectiveness of the proposed method. Section 4 gives some concluding remarks.

## 2 Chebyshev Collocation Method

In this section we derive the Chebyshev-collocation method to problem (1)-(3). Before this we will introduce some basic properties of the most commonly used set of orthogonal polynomials; Chebyshev polynomials.

### 2.1 Chebyshev Polynomials

The Chebyshev polynomials $\{T_n(x)\}$, $n = 0, 1, \ldots,$ are the Eigen functions of the singular Sturm-Liouville problem

$$\frac{d}{dx}\left(\sqrt{1-x^2}\,\frac{dT_n(x)}{dx}\right) + \frac{n^2}{\sqrt{1-x^2}}T_n(x) = 0,$$

they are also orthogonal with respect to $L_2$ inner product on the interval [-1, 1] with the weight function $\omega(x) = (1-x^2)^{-\frac{1}{2}}$

$$\int_{-1}^{1} T_j(x)T_k(x)\omega(x)dx = \frac{c_j\pi}{2}\delta_{jk},$$

where $\delta_{jk}$ is the Kronecker delta, $c_0 = 2$ and $c_j = 1$ for $j \geq 1$.

The Chebyshev polynomials satisfy the following three-term recurrence relation

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \quad n \geq 1,$$
$$T_0(x) = 1, \quad T_1(x) = x,$$

and

$$2T_n(x) = \frac{1}{n+1}T'_{n+1}(x) - \frac{1}{n-1}T'_{n-1}(x), \quad n \geq 2, \tag{4}$$
$$T_0(x) = T'_1(x), \quad 2T_1(x) = 0.5T'_2(x).$$

A unique feature of the Chebyshev polynomials is their explicit relation with a trigonometric function:

$$T_n(x) = \cos(n \arccos x). \tag{5}$$

### 2.2 Chebyshev Collocation Approximation

Now, consider problem (1)-(3) on domain Ω. Because of the orthogonally property of the Chebyshev polynomials on the interval [-1, 1], we transfer (1) from [0, $T$] to an equivalent problem defined in [-1, 1], by using the substitution

$$t = \frac{T}{2}(\tau+1), \quad \tau \in [-1,1],$$

then (1) will convert to

$$\frac{2}{T}\frac{\partial U}{\partial \tau} - \frac{\partial^2 U}{\partial x^2} = \int_0^{\frac{T}{2}(\tau+1)} k\left(x, \frac{T}{2}(\tau+1) - s\right)\Delta u(x,s)ds + g(x,\tau), \tag{6}$$

in which

$$U := U(x,\tau) = u\left(x, \frac{T}{2}(\tau+1)\right), \quad g(x,\tau) = f\left(x, \frac{T}{2}(\tau+1)\right),$$

for all $x, \tau \in \Omega$.

By using the following change of variable

$$s = \frac{T}{2}(\rho+1), \quad \rho \in [-1,1],$$

we convert the integration interval from $[0, \frac{T}{2}(\tau+1)]$ to $[-1, \tau]$ in (6), so that (6) will be

$$\frac{2}{T}\frac{\partial U}{\partial \tau} - \frac{\partial^2 U}{\partial x^2} = \int_{-1}^{\tau} K(x, \tau-\rho)\Delta U(x,\rho)d\rho + g(x,\tau), \tag{7}$$

where $K(x, \tau-\rho) = k(x, \frac{T}{2}(\tau-\rho))$. Define the auxiliary function

$$\Phi(x,\tau) = \frac{T}{2}\int_{-1}^{\tau} K(x,\tau-\rho)U_{xx}(x,\rho)d\rho + g(x,\tau). \tag{8}$$

In order to approximate problem (1)-(3) by spectral methods, we restart (7) as two equivalent Volterra Integro-differential equations by using (8) as the following

$$\begin{cases} \dfrac{2}{T}\dfrac{\partial U}{\partial \tau} - \dfrac{\partial^2 U}{\partial x^2} = \Phi(x,\tau), \\ \Phi(x,\tau) = \dfrac{T}{2}\displaystyle\int_{-1}^{\tau} K(x,\tau-\rho)U_{xx}(x,\rho)d\rho + g(x,\tau), \end{cases} \tag{9}$$

by integration of both sides of the first part in (9) over the interval [-1, $\tau$], we get

$$\begin{cases} U(x,\tau) = u_0(x) + \dfrac{T}{2}\displaystyle\int_{-1}^{\tau}[\Phi(x,\xi) + U_{xx}(x,\xi)]d\xi, \\ \Phi(x,\tau) = \dfrac{T}{2}\displaystyle\int_{-1}^{\tau} K(x,\tau-\rho)U_{xx}(x,\rho)d\rho + g(x,\tau). \end{cases} \tag{10}$$

Let the collocation points be the set of $(N+1)(M+1)$ points $\{(x_i,\tau_j)\}$ in which $\{x_i = -\cos(\frac{\pi}{N}i)\}_{i=0}^{N}$ are the Chebyshev-Gauss-Lobatto nodes (CGL nodes) and $\tau_j$ are the Chebyshev-Gauss nodes (CG nodes) defined as $\{\tau_j = -\cos(\frac{2j+1}{2M+2}\pi)\}_{j=0}^{M}$.

Equation (10) holds at $\{(x_i,\tau_j)\}$

$$\begin{cases} U(x_i,\tau_j) = u_0(x_i) + \dfrac{T}{2}\displaystyle\int_{-1}^{\tau_j}[\Phi(x_i,\xi) + U_{xx}(x_i,\xi)]d\xi, \\ \qquad\qquad 1 \leq i \leq N-1, \quad 0 \leq j \leq M, \\ \Phi(x_i,\tau_j) = \dfrac{T}{2}\displaystyle\int_{-1}^{\tau_j} K(x_i,\tau_j-\rho)U_{xx}(x_i,\rho)d\rho + g(x_i,\tau_j), \\ \qquad\qquad 0 \leq i \leq N, \quad 0 \leq j \leq M, \end{cases} \tag{11}$$

For approximating the integral terms in (11) the integral interval will transfer from [-1, $\tau_j$] to a fixed one [-1, 1] by using a simple linear transformation

$$\rho(\tau_j,\theta) = \xi(\tau_j,\theta) = \frac{\tau_j+1}{2}\theta + \frac{\tau_j-1}{2}, \quad \theta \in [-1,1],$$

where $\{\theta_k\}_{k=0}^{p}$ are roots of the $(p+1)$-th Chebyshev polynomials. Then (11) becomes:

$$\begin{cases} U(x_i,\tau_j) = u_0(x_i) + \\ \qquad .\dfrac{T(\tau_j+1)}{4}\displaystyle\int_{-1}^{1}[\Phi(x_i,\xi(\tau_j,\theta)) + U_{xx}(x_i,\xi(\tau_j,\theta))]d\theta, \\ \Phi(x_i,\tau_j) = g(x_i,\tau_j) + \\ \qquad .\dfrac{T(\tau_j+1)}{4}\displaystyle\int_{-1}^{1} K(x_i,\tau_j-\rho(\tau_j,\theta))U_{xx}(x_i,\rho(\tau_j,\theta))d\theta, \end{cases} \tag{12}$$

Now, we use

$$\begin{cases}\Phi_N^M(x,\sigma) := \sum_{n=0}^{N}\sum_{m=0}^{M} l_n(x)F_m(\sigma)\varphi(x_n,\tau_m),\\ U_N^M(x,\sigma) := \sum_{n=0}^{N}\sum_{m=0}^{M} l_n(x)F_m(\sigma)U(x_n,\tau_m),\end{cases} \tag{13}$$

to approximate the functions Φ and U, where $l_n(x)$ and $F_m(\sigma)$ are the n-th and m-th Lagrange basis functions corresponding to non-uniform meshes of $\{x_i\}$ and $\{\tau_j\}$ respectively. After enforcing the homogeneous boundary conditions at $x_0 = -1$ and $x_N = 1$ the first and the last terms in the interpolation polynomial of U are omitted. Therefore, we have

$$U_N^M(x,\sigma) := \sum_{n=1}^{N-1}\sum_{m=0}^{M} l_n(x)F_m(\sigma)U(x_n,\tau_m). \tag{14}$$

Now we can approximate $U_{xx}$ in (12) by using the interpolation polynomial of $U_N^M$ from the previous equation as follows

$$(U_N^M)_{xx}(x,\sigma) := \sum_{n=1}^{N-1}\sum_{m=0}^{M} l_n''(x)F_m(\sigma)U(x_n,\tau_m). \tag{15}$$

Where $l_n''(x) = D^2$ is the second derivative of the Lagrange interpolation function $l_n(x)$ which is a polynomial of degree N-2, which can be defined as introduced in [24], [25], and [26] the second derivative of the differentiation matrix $D_{N+1}$. Now, we write the entries of $D^2 = [D_{i,k}^2]$ for $\{x_i\}$ as the following

$$D_{i,k}^2 = \begin{cases} 2D_{i,k}\left(D_{i,i} - \dfrac{1}{x_i - x_k}\right), & i \neq k,\\ -\sum_{l=0, i\neq 1}^{N} D_{i,1}^2, & i = k,\end{cases}$$

where $D_{i,k}$ the entries of the so-called differentiation matrix, which has a dimension of $(N+1)$. The entries of the differentiation matrix can be defined in [27] for (CG) points as the following

$$D_{i,k} = \begin{cases} -\dfrac{2N^2+1}{6}, & i = k = 0,\\ \dfrac{c_i}{c_k}\dfrac{(-1)^{i+k}}{x_i - x_k}, & i \neq k,\\ -\dfrac{x_i}{2\left(1 - x_i^2\right)}, & 1 \le i = k \le N-1,\\ \dfrac{2N^2+1}{6}, & i = k = N,\end{cases}$$

with $c_i = 2$ *for* i = 0, N, and $c_i = 1$ otherwise. Now we approximate the integration in (12) by replacing the integral function by its interpolation polynomial approximation of Φ and approximation of $U_{xx}$ from (13) and (15) respectively in (12), and writing, $\Phi(x_i,\tau_j) = \Phi_{i,j}, U(x_i,\tau_j) = U_{i,j}, g(x_i,\tau_j) = g_{i,j}$. Then our goal is to find $U_{i,j}$ so we obtain the following

$$\begin{cases} U_{i,j} = u_0(x_i) + \dfrac{T(\tau_j+1)}{4} \cdot \int_{-1}^{1} I_N^M[\Phi_N^M(x_i,\xi(\tau_j,\theta)) + (U_N^M)_{xx}(x_i,\xi(\tau_j,\theta))]d\theta,\\ \Phi_{i,j} = g_{i,j} + \dfrac{T(\tau_j+1)}{4} \cdot \int_{-1}^{1} I_N^M[K(x_i,\tau_j - \rho(\tau_j,\theta))(U_N^M)_{xx}(x_i,\rho(\tau_j,\theta))]d\theta,\end{cases} \tag{16}$$

where $I_N^M$ is the interpolation operator associated with the Chebyshev mesh points $\{(x_i, \tau_j)\}$, defined as the following;

$$I_N^M Q := \sum_{n=1}^{N-1}\sum_{m=0}^{M} l_n(x)F_m(\sigma)Q(x_n,\tau_m).$$

Now each equation in (16) can be reformulated respectively as

$$\Phi_{i,j} = g_{i,j} + \frac{T(\tau_j+1)}{4}\sum_{n=1}^{N-1}\sum_{m=0}^{M} l_n''(x_i)U(x_n,\tau_m) \cdot \sum_{z=1}^{p-1} l_z(x_i)\sum_{k=0}^{p} K(x_i,\tau_j - \rho(\tau_j,\theta_k))F_m(\rho(\tau_j,\theta_k))\int_{-1}^{1} F_k(\theta)d\theta, \tag{17}$$

$$U_{i,j} = u_0(x_i) + \frac{T(\tau_j+1)}{4}\Big[\sum_{n=0}^{N}\sum_{m=0}^{M} l_n(x_i)\Phi(x_n,\tau_m) + \sum_{n=1}^{N-1}\sum_{m=0}^{M} l_n''(x_i)U(x_n,\tau_m)\Big] \cdot \sum_{z=1}^{p-1} l_z(x_i)\sum_{k=0}^{p} F_m(\xi(\tau_j,\theta_k))\int_{-1}^{1} F_k(\theta)d\theta, \tag{18}$$

Now we discuss an efficient way to find $d_k = \int_{-1}^{1} F_k(\theta)d\theta$. First we express $F_j(s)$ in terms of Chebyshev functions as in [20]:

$$F_j(s) = \omega_j^C \sum_{p=0}^{N} (T_p(x_j)/\gamma_p)T_p(s), \tag{19}$$

where $\omega_j^C$ is the Chebyshev weight corresponding to Chebyshev points $\{x_i\}_{i=0}^N$ and

$$\gamma_p = \sum_{i=1}^{N} T_p^2(x_i)\omega_i^C = \begin{cases}\pi & p = 0,\\ \pi/2, & 1 \le p < N,\end{cases} \tag{20}$$

and $\gamma_N = \pi/2$ if $\{x_i\}_{i=0}^N$ is the Chebyshev Gauss or the Chebyshev Gauss Radau points, where if we use $\{x_i\}_{i=0}^N$ as the Chebyshev Gauss Lobatto then $\gamma_N = \pi$. From (19) we can now calculate $d_k = \int_{-1}^{1} F_k(\theta)d\theta$ as following

$$d_k = \omega_k^C \sum_{p=0}^{N} (T_p(x_k)/\gamma_p)\int_{-1}^{1} T_p(\theta)d\theta.$$

To compute $\int_{-1}^{1} T_p(\theta)d\theta$, we use the recurrence relation (4) for Chebyshev polynomials yields

$$\int_{-1}^{1} T_p(\theta)d\theta = \begin{cases}\dfrac{2}{1-p^2}, & \text{p is even number,}\\ 0, & \text{othrwise.}\end{cases}$$

Now rewrite (17) and (18) in the matrix form as

$$\begin{cases} \mathbf{\Phi}_{(N+1)(M+1)} = \mathbf{G}_{(N+1)(M+)} + L\mathbf{U}_{(N-1)(M+1)}, \\ \mathbf{U}_{(N-1)(M+1)} = \mathbf{U}_{-1} + A\mathbf{\Phi}_{(N+1)(M+1)} + B\mathbf{U}_{(N-1)(M+1)}, \end{cases}$$

where $\mathbf{\Phi}, \mathbf{G}, \mathbf{U}$ *and* $\mathbf{U}_{-1}$ represent vectors, each one defined as the following

$$\begin{cases} \mathbf{\Phi}_{(N+1)(M+1)} = vec\left[\Phi_{i,j}\right], \quad \mathbf{G}_{(N+1)(M+)} = vec\left[g_{i,j}\right], \quad 0 \le i \le N, \\ \mathbf{U}_{(N-1)(M+1)} = vec\left[U_{i,j}\right], \quad 1 \le i \le N-1, \quad 0 \le j \le M, \end{cases}$$

$$\mathbf{U}_{-1} = vec\begin{bmatrix} u_0(x_2) & u_0(x_3) & \cdots & u_0(x_{N-1}) \\ \vdots & \vdots & \vdots & \vdots \\ u_0(x_2) & u_0(x_3) & \cdots & u_0(x_{N-1}) \end{bmatrix},$$

in which the *vec* operator reshapes any matrix into a vector by placing columns of the matrix below each other from the first to the last. For the other matrices each one can defined as block ones as the following:

- $A = (A_j^i)$, A is a matrix with dimension of $(N+1)(N-1) \times (N+1)^2$ in which the first and last *(*N+*1)* columns are zeros and the other blocks $(A_j^i)$ forms a diagonal matrix in which its entries are given by

$$\left(A_j^i\right)_{i,i+1} = \frac{T(\tau_j+1)}{4}\sum_{k=0}^{p} d_k F_m(\xi(\tau_j,\theta_k)),$$

with *(*$0 \le m \le M$, $0 \le j \le M$, $1 \le i \le N-1$, *and* $1 \le n \le N-1$*).* For each block in $(A_i^j)$ we obtain a matrix with dimension of $(N+1)\times(N+1)$. The shape of the global matrix A will be

$$A = \begin{bmatrix} \mathbf{0} & A_j^1 & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & A_j^{N-1} & \mathbf{0} \end{bmatrix}$$

- $L = \left(L_j^i\right)$, $L$ is a matrix with dimension $(N+1)^2 \times (N+1)(N-1)$ in which the first and last *(*N+*1)* rows are zeros, and each block matrix $(L_j^i)$ has a dimension $(N+1)\times(N+1)$. The entries of $(L_j^i)$ can give by the following

$$\left(L_j^i\right)_{i+1,n} = \frac{T(\tau_j+1)}{4}\sum_{k=0}^{p} d_k K(x_i,\tau_j - \rho(\tau_j,\theta_k))F_m(\rho(\tau_j,\theta_k))D_{i,n}^2,$$

with ($0 \le m \le M$, $0 \le j \le M$, $1 \le i \le N-1$, *and* $1 \le n \le N-1$) respectively.

- *Finally* the matrix $B = \left(B_j^i\right)$, $B$ has the dimension $(N+1)(N-1)\times(N+1)(N-1)$. *The entries of matrices* $(B_j^i)$ *is* obtained as the following

$$\left(B_j^i\right)_{i,n} = \frac{T(\tau_j+1)}{4}\sum_{k=0}^{p} d_k F_m(\xi(\tau_j,\theta_k))D_{i,n}^2,$$

with *(*$0 \le m \le M$, $0 \le j \le M$, $1 \le i \le N-1$, *and* $1 \le n \le N-1$*)* respectively. Each matrix in the previous equation of dimension $(N+1)\times(N+1)$.

For each previous matrix we can use kronecker product to find them well.

To solve the coupled equations system in (17) and (18), we convert them to a linear algebraic system as follows

$$(I - B - AL)\mathbf{U}_{(N-1)(M+1)} = \mathbf{U}_{-1} + A\mathbf{G}_{(N+1)(M+1)}.$$

After solving the previous system, we obtain an approximation to $\mathbf{U}_{(N-1)(M+1)}$, then the approximation to the original problem for all $x \in (-1, 1)$, and $t \in [0, T]$ can found by

$$u(x,t) \approx \sum_{n=1}^{N-1}\sum_{m=0}^{M} l_n(x)F_m\left(\frac{2}{T}t-1\right)U_{n,m}.$$

## 3 Numerical Results

In order to test the utility of the proposed new method, we devoted this section to some numerical examples to view the efficiency and accuracy of the method in the previous one. In our implementation , we set $T = 1$, $p = N = M$, and let $\{\theta_k\}_{k=0}^{p}$ be the Chebyshev-Gauss points with the corresponding weights $\omega_k^C = \pi/(N+1)$. To show the efficiency of the previous method for our problems in comparison with the exact solution, we calculate for different values of $N$ the maximum error defined by

$$\|E\|_\infty = \max_{\substack{1\le i\le N-1 \\ 0\le j\le M}} \left| U_{i,j} - u\left(x_i, \frac{T(\tau_j+1)}{2}\right)\right|.$$

**Example 3.1:** Consider the linear problem (1)-(3) with the kernel $k(x, t) = exp(-x^2 t)$ and we choose the forcing function $f(x, t)$ so that $u(x, t) = (1 - x^2)\, exp(t)$ is the exact solution.

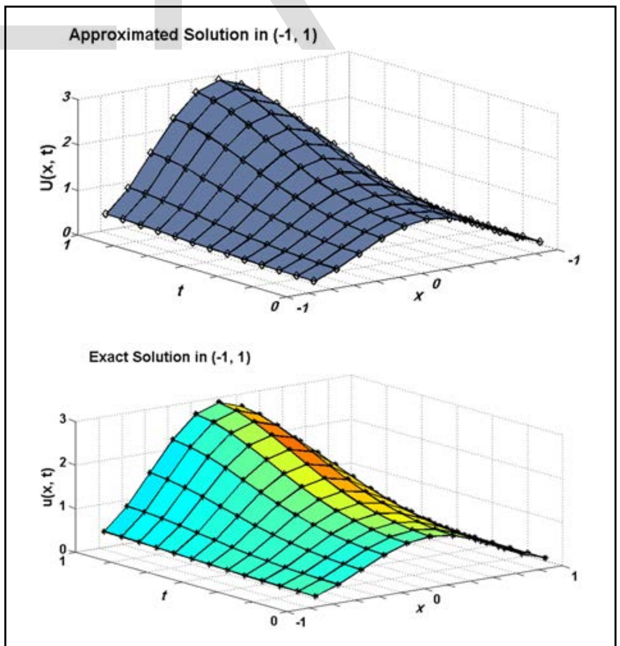


Fig. 1. Approximated and Exact solutions respectively for $x \in (-1, 1)$ and $t \in [0, 1]$ at $N = 12$.

TABLE 1
MAXIMUM $L^\infty$ ERRORS FOR EXAMPLE (3.1)

| $N = M$ | $\| \mathbf{E} \|_\infty$ | CPU time(s) |
|---|---|---|
| 4 | 3.0122E-06 | 1.386961 |
| 6 | 3.4863E-09 | 3.356867 |
| 8 | 2.0506E-12 | 7.304112 |
| 10 | 1.4211E-14 | 13.765722 |
| 12 | 3.3084E-14 | 22.416617 |
| 14 | 4.9738E-14 | 36.239783 |

CPU time(s) indicates the time for all calculations for the program from start to the end.

**Example 3.2:** Consider the linear problem (1)-(3) with the kernel $k(x, t) = exp\ (-x^2 t)$ and we choose the forcing function $f(x, t)$ so that $u(x, t) = (1- x^4)\ exp\ (x+t)$ is the exact solution.

TABLE 2
MAXIMUM $L^\infty$ ERRORS FOR EXAMPLE (3.2)

| $N = M$ | $\| \mathbf{E} \|_\infty$ | CPU time(s) |
|---|---|---|
| 6 | 6.6138E-03 | 3.436899 |
| 8 | 5.7533E-05 | 8.520456 |
| 10 | 2.6170E-07 | 14.49634 |
| 12 | 7.3104E-10 | 23.69198 |
| 14 | 1.3947E-12 | 41.66479 |
| 16 | 8.5043E-14 | 73.698840 |

CPU time(s) indicates the time for all calculations for the program from start to the end.

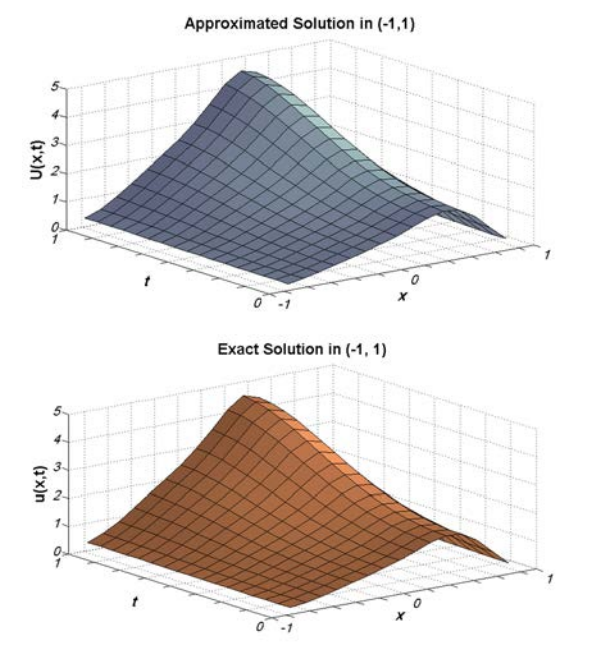


Fig. 2. Approximated and Exact solutions respectively for $x \in (-1, 1)$ and $t \in [0, 1]$ at $N = 16$.

It can be seen that the errors in Tables (1)-(2) decay rapidly, which is confirmed by spectral accuracy.
All the computations are carried out in double precision arithmetic using Matlab 7.9.0 (R2009b). To obtain sufficient accurate calculations, variable arithmetic precision (vpa) is employed with digit being assigned to be 32. The code was executed on a second generation Intel Core i5-2410M, 2.3 Ghz Laptop.

## 4 CONCLUSION

This article used a new competitive numerical scheme based on developing Chebyshev spectral collocation method to find the approximate solution of parabolic Volterra integro-differential equations. The main advantage of using Chebyshev scheme instead of using Legendre one is that its quadrature points have explicit and simple expressions as well as the corresponding weights. This enables us to avoid the complex computation of the Legendre quadrature points and the corresponding weights. Moreover, we made a minor modification on approximating the integration by replacing the integral function by its interpolating polynomials instead of using Gauss quadrature approximation and this increases the accuracy of the suggested method. The numerical examples given in this work have demonstrated the potential of the newly proposed numerical scheme in solving parabolic Volterra integro-differential and similar equations even with using a small number of collocation points.

## REFERENCES


[1] E. G. Yanik, and G. Fairweather, "Finite element methods for parabolic and hyperbolic partial integro-differential equations", Nonlinear Anal. (1988), 12, pp. 785–809.

[2] M. Renardy, W. Hrusa, and J. Nohel, "Mathematical Problems in Viscoelasticity", Pitman monographs and Surveys in pure Appl. Math., New York: wiley. (1987), 35.

[3] T. A. Burton, "*Volterra Integral and Differential Equations*", (second edition), Mathematics in Science and Engineering. Elsevier, vol. 202, 2005.

[4] G. Dagan, "The significance of heterogeneity of evolving scales to transport in porous formations", Water Resour. Res., (1994), 30, pp. 3327-3337.

[5] J. Cushman, and T. Glinn, "Nonlocal dispersion in media with continuously evolving scales of heterogeneity", Transport in Porous Media, (1993), 13, pp. 123-138.

[6] R. Ewing, Y. Lin, and J. Wang, "A numerical approximation of non-Fickian flows with mixing length growth in porous media", Acta. Math. Univ. comenian. (2001), LXX, pp. 75-84.

[7] Amiya K. Pani, and G. Fairweather, "$H^1$-Galerkin mixed finite element methods for parabolic partial integro-differential equations", IMA Journal of Numerical Analysis. (2002), 22, pp. 231-252.

[8] Amiya K. Pani, G. Fairweather, and Ryan I. Fernandes, "ALTERNATING DIRECTION IMPLICIT ORTHOGONAL SPLINE COLLOCATION METHODS FOR AN EVOLUTION EQUATION WITH A POSITIVE-TYPE MEMORY TERM", SIAM J. NUMER. ANAL. (2008), 46, pp. 344-364.

[9] Amiya K. Pani, G. Fairweather, and Ryan I. Fernandes, "ADI orthogonal spline collocation methods for parabolic partial integro-differential equations", IMA Journal of Numerical Analysis. (2010), 30, pp. 248-276.

[10] F. Fakhar-Izadi, and M. Dehghan, "The spectral methods for parabolic Volterra integro-differential equations", Journal of Computational and Applied Mathematics. (2011), 235, pp. 4032-4046.

[11] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, "*Spectral Methods: Fundamentals in Single Domains*", Springer-Verlag, 2006.

[12] L. N. Trefethen, "*Spectral Methods in MATLAB*", Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.

[13] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, "*Spectral Methods in Fluid Dynamics*", Springer-Verlag, 1988.

[14] W. Guo, G. Labrosse, and R. Narayanan, "*The Application of the Chebyshev-Spectral Method in Transport Phenomena*", Notes in Applied and Computational Mechanics 68, Springer-Verlag Berlin Heidelberg, 2012.

[15] J. P. Boyd, "*Chebyshev and Fourier Spectral Methods*", Dover Publications, Mineola, NY, 2001.

[16] J. Shen, and T. Tang, "*Spectral and High-Order Methods with applications*", Science Press, Beijing, China. 2006.

[17] M. Aguilar, and H. Brunner," Collocation methods for second-order Volterra integro-differential equations", Appl. Numer. Math. (1988), 4, pp. 455--470.

[18] D. Conte, and B. Paternoster, "Multistep collocation methods for Volterra Integral Equations", Applied Numerical Mathematics. (2009), 59, pp. 1721-1736.

[19] Tobin A. Driscoll, "Automatic spectral collocation for integral, integro-differential, and integrally reformulated differential equations", Journal of Computational Physics. (2010), 229, pp. 5980-5998.

[20] T. Tang, "On spectral methods for Volterra integral equations and the convergence analysis", J. Comput. Math. (2008), 26, pp. 825-837.

[21] Z. Wan, Y. Chen, and Y. Huang, "Legendre spectral Galerkin method for second-kind Volterra integral equations", Front. Math. China. (2009), 4, pp. 181-193.

[22] Zhendong Cu, and Yanping Chen, "Chebyshev spectral-collocation method for Volterra integral equations", Recent Advances in Scientific Computing and Applications, Contemporary Mathematics. (2013), 586, pp. 163-170.

[23] Wu Hua, and Zhang Jue, "Chebyshev-Collocation Spectral Method for Volterra Integro-Differential equations", Journal of Shanghhai University (Natural Scince). (2011), 17, pp. 182-188.

[24] B. Costa, and W. S. Don, "On the computation of high order pseudo-spectral derivatives", Appl. Numer. Math. (2000), 33, pp. 151-159.

[25] R. Baltensperger, and M. R. Trummer, "Spectral Differencing with a twist", J. Indus. and Appl. Math. (2003), 24, pp. 1465--1487.

[26] Elsayed M. E. Elbarbary, and Salah M. El-Sayed, "Higher order pseudospectral differentiation matrices", Appl. Numer. Math. (2005), 55, pp. 425-438.

[27] J. S. Hesthaven, S. Gottlieb, and D. Gottlieb, "*Spectral Methods for Time-Dependent Problems*", Cambridge University Press. 21, pp. 96-99, 2007.