fractal and fractional

Article

An Efficient Cubic B-Spline Technique for Solving the Time Fractional Coupled Viscous Burgers Equation

Usama Ghafoor [1], Muhammad Abbas [1,*], Tayyaba Akram [2], Emad K. El-Shewy [3,4], Mahmoud A. E. Abdelrahman [5,6] and Noura F. Abdo [5,6]

1 Department of Mathematics, University of Sargodha, Sargodha 40100, Pakistan
2 Center of Excellence in Theoretical and Computational Science (TaCS-CoS), Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha-Uthit Road, Bang Mod, Thung Khru, Bangkok 10140, Thailand
3 Department of Physics, College of Science, Taibah University, Al-Madinah Al-Munawarah 41477, Saudi Arabia
4 Theoretical Physics Group, Faculty of Science, Mansoura University, Mansoura 35516, Egypt
5 Department of Mathematics, College of Science, Taibah University, Al-Madinah Al-Munawarah 41477, Saudi Arabia
6 Department of Mathematics, Faculty of Science, Mansoura University, Mansoura 35516, Egypt
* Correspondence: muhammad.abbas@uos.edu.pk

Abstract: The second order Burger's equation model is used to study the turbulent fluids, suspensions, shock waves, and the propagation of shallow water waves. In the present research, we investigate a numerical solution to the time fractional coupled-Burgers equation (TFCBE) using Crank–Nicolson and the cubic B-spline (CBS) approaches. The time derivative is addressed using Caputo's formula, while the CBS technique with the help of a θ-weighted scheme is utilized to discretize the first- and second-order spatial derivatives. The quasi-linearization technique is used to linearize the non-linear terms. The suggested scheme demonstrates unconditionally stable. Some numerical tests are utilized to evaluate the accuracy and feasibility of the current technique.

Keywords: cubic B-spline; coupled Burgers equation; Crank–Nicolson finite difference technique; Caputo derivative; quasi-linearization

MSC: 39A12; 39B62; 33B10; 26A48; 26A51

check for updates

Citation: Ghafoor, U.; Abbas, M.; Akram, T.; El-Shewy, E.K.; Abdelrahman, M.A.E.; Abdo, N.F. An Efficient Cubic B-Spline Technique for Solving the Time Fractional Coupled Viscous Burgers Equation. *Fractal Fract.* **2024**, *8*, 93. https://doi.org/10.3390/fractalfract8020093

Academic Editor: Carlo Cattani

Received: 7 September 2023
Revised: 8 January 2024
Accepted: 20 January 2024
Published: 31 January 2024

1. Introduction

The second order viscous Coupled Burgers equations (VCBEs) are considered for numerical solution in this article. This model is used to study physical phenomena such as turbulent fluids, fluid suspensions, shock waves, arising in gas dynamics, the propagation of shallow water waves, continuous stochastic processes, flow theory, and chromatography [1–5]. The non-linear VCBEs are defined as follows:

$$\begin{cases} U_t(x,t) - U_{xx}(x,t) + \alpha_1(U(x,t)V(x,t))_x + \gamma U(x,t)U_x(x,t) = 0, & a \le x \le b, \quad t \in [0,T], \\ V_t(x,t) - V_{xx}(x,t) + \alpha_2(V(x,t)U(x,t))_x + \gamma V(x,t)V_x(x,t) = 0, & a \le x \le b, \quad t \in [0,T], \end{cases} \tag{1}$$

with initial conditions (ICs)

$$U(x,0) = \phi_1(x), \quad V(x,0) = \phi_2(x), \quad a \le x \le b \tag{2}$$

and boundary conditions (BCs)

$$\begin{cases} U(a,t) &= \varphi_1(t), V(a,t) = \psi_1(t), \\ U(b,t) &= \varphi_2(t), V(b,t) = \psi_2(t), \end{cases} \quad t \in [0,T] \tag{3}$$

where α_1 and α_2 are system parameters, which represent the diffusion and interaction of two fluids, respectively, and γ is a constant.

Fractional calculus involves the study of non-integer order derivatives and integrals [6–8]. The nonlinear TFCBE with different parameters is defined as follows:

$$\begin{cases} \frac{\partial^\lambda U(x,t)}{\partial t^\lambda} - \frac{\partial^2 U(x,t)}{\partial x^2} + \alpha_1 \frac{\partial (U(x,t)V(x,t))}{\partial x} + \gamma U(x,t)\frac{\partial U(x,t)}{\partial x} = Q_1, \\ \frac{\partial^\lambda V(x,t)}{\partial t^\lambda} - \frac{\partial^2 V(x,t)}{\partial x^2} + \alpha_2 \frac{\partial (U(x,t)V(x,t))}{\partial x} + \gamma V(x,t)\frac{\partial V(x,t)}{\partial x} = Q_2, \end{cases} x \in [a,b], t \in [0,T], \lambda \in (0,1), \tag{4}$$

with ICs:

$$U(x,0) = \phi_1(x), \;\; V(x,0) = \phi_2(x), \tag{5}$$

the BCs:

$$\begin{cases} U(a,t) = \varphi_1(t), & U(b,t) = \varphi_2(t), \\ V(a,t) = \psi_1(t), & V(b,t) = \psi_2(t), \end{cases} \tag{6}$$

where λ is the fractional parameter, $\frac{\partial^\lambda U(x,t)}{\partial t^\lambda}$ is the time fractional derivative (FD) in the Caputo sense, and $\frac{\partial U(x,t)}{\partial x} = U_x(x,t)$ and $\frac{\partial^2 U(x,t)}{\partial t^2} = U_{xx}(x,t)$ are first- and second-order spatial derivatives, respectively. This model represents the time fractional coupled-Burgers equation (TFCBE) in Caputo's sense that satisfies the mathematical model accurately and represents the real-world system. These equations are well defined and capture the memory in time direction. The velocity profiles of two distinct fluids are denoted as $U(x,t)$ and $V(x,t)$. There are numerous methods to deal with the FD. Usually, the Caputo FD is used to acquire the appropriate real world physical models. In this paper, Caputo's FD of the function is employed to discretize the temporal derivative. The Caputo FD of the function $U(x,t)$ with order $\lambda \in (0,1)$ is given as follows:

$$\frac{\partial^\lambda U(x,t)}{\partial t^\lambda} = \begin{cases} \frac{1}{\Gamma(1-\lambda)} \int_0^t \frac{\partial U(x,\kappa)}{\partial \kappa} \frac{d\kappa}{(t-\kappa)^\lambda}, & 0 < \lambda < 1, \\ \frac{\partial U(x,t)}{\partial t}, & \lambda = 1, \end{cases} \tag{7}$$

where the Euler's Gamma function is represented by Γ.

Numerous approaches have been developed to deal with the initial and boundary value problems. Chen et al. [9] represented an approximate solution to coupled Burgers equations (CBEs) with FD using the Adomian decomposition method (ADM). Khan et al. [10] studied the Burgers and a system of Burgers equations using the homotopy perturbation method (HPM). They also solved these problems using a generalized differential transform method (GDTM) involving the Caputo time FD. CBEs have been solved by Prakash et al. [11] via a time fractional variational iteration approach. They compared their results with other methods, namely the HPM, the ADM, and the generalized differential transformation method (GDTM). The CBEs have also been solved using the q-homotopy analysis transform method by Singh et al. [12]. Aminikhah and Malekzadeh [13] developed a new HPM, in order to calculate the system of the CBEs and temporal FD that is used with the Caputo formula. In [14], a reduced differential transform method, the Laplace ADM, and the Laplace-variational iteration scheme have been utilized to investigate a solution to the CBEs. Albuohimad and Adibi proposed a solution to the CBEs using the hybrid spectral exponential Chebyshev method (HSECM), and the FD derivative was discretized utilizing the finite difference method (FDM) [15]. Sulaiman investigated the system of viscous Burgers equations using the Atangana–Baleanu FD to obtain a numerical simulation in [16]. A system Burgers equation was estimated using the generalized differential transform method (DTM) and the Caputo derivative (CD), employed to solve the temporal FD in [17]. Ozdemir et al. [18] calculated approximate solutions to Burgers equations by proposing Gegenbauer wavelets-based computational methods with TFD. Abazari and Borhanifar [19] developed an approximate solution to the CBEs through a differential transformation method. They also compared their approximate results with three different methods, namely the variational iteration, HPM, and analysis techniques.

Mittal and Arora [20] investigated a system of viscous Burgers equations, discretized the scheme using the CBS collocation, and investigated the stability of the scheme. Shukla et al. [21] used a modified CBS to study the two-dimensional nonlinear VCBEs and found approximate solutions using the differential quadrature method. Kumar and Arora [22] employed approximate solutions to the coupled Klein Gordon equation and a system of Burgers equations through a reduced differential transformation scheme. Srivastava et al. [23] used the implication logarithmic FDM to present an approximation of a solution to the nonlinear CBEs. Mittal and Tripathi [24] investigated an approximate solution to a system of Burgers equation with the help of a modified spline. Sarboland and Aminataei [25] solved the nonlinear CBEs through a multiquadric quasi-interpolation scheme. Salih et al. [26] proposed a numerical solution to CBEs with Dirichlet's boundary conditions by using trigonometric spline functions. He and Tang [27] developed a lattice Boltzmann to find a numerical solution to CBEs. Chuathong and Kaennakham [28] suggested a collocation method to determine approximate solutions to CBEs. Jima et al. [29] presented an approximated solution using a differential quadrature method. In [30], the authors presented a Chebyshev wavelets method to investigate a CBE. Guo-Cheng et al. [31] discussed the concept of short memory fractional differential equations with impulses and provided exact solutions for linear cases. Dubey et al. [32] presented a computational algorithm called a local fractional natural homotopy analysis method to solve local fractional coupled Helmholtz and CBEs in fractal media. They also discussed the uniqueness, convergence, and error analysis of the solutions obtained using the method. Numerical simulations on the Cantor set validate the effectiveness of the proposed method. WANG [33] proposed a new fractal modified equal width-Burgers equation (MEWBE) with the local fractional derivative using a Mittag–Leffler function.

Spline interpolation is a valuable method for approximating mathematical functions with a piecewise smooth curve. Specifically, B-spline interpolation utilizes a type of piecewise polynomial that exhibits high localization. There is no study on the implementation of cubic B-spline for the TFCBEs. Inspired by the success of the spline in the numerical solution to integer-order differential equations and fractional order differential equations, the main aim of this study is to investigate the numerical solution to TFCBEs by using cubic B-spline functions. They have been employed by several researchers for solving fractional partial differential equations. The novelty of the proposed work is to discretize the first-order time fractional derivative in the Caputo sense with a first-order Crank–Nicholson finite difference scheme, while the CBS functions are used to discretize the spatial derivatives for coupled equations. It can be easily seen that the work done in this paper provides a more accurate numerical solution to the proposed problem because we use a combination of a Crank–Nicholson finite difference scheme and CBS functions with the help of a θ-weighted scheme. This first-order time fractional derivative discretization in the Caputo sense for coupled equations has never been used, as far as we are aware, for the case of first-order TFCBEs. The superiority of the proposed method is to provide a numerical solution in a piecewise cubic function with the smoothness of C^2 continuity at each joint point of solution.

The paper is structured as follows: Section 2 describes the CBS functions at different knots. Sections 3 and 4 present the discretization of the fractional derivative and the procedure of the proposed numerical technique, respectively. Section 5 discusses the stability of the suggested technique. The conclusion of the study is presented in Section 6.

2. B-Spline Approximation

Let $a = x_0 < x_1 ... x_{N-1} < x_N = b$ be the spatial domain on the interval $[a, b]$. In the present section, cubic B-spline approximation for $U(x,t)$ and $V(x,t)$ is constructed. We define the partition into equivalent sub-intervals of length $h = \frac{b-a}{N}$, where $x_i = a + ih$, where $i = 0, 1, 2, 3, ..., N$. The cubic B-spline can be defined as follows:

$$B_i(x) = \frac{1}{6h^3}\begin{cases}(x - x_{i-2})^3, & x \in [x_{i-2}, x_{i-1}],\\ h^3 - 3(x - x_{i-1})^3 + 3h(x - x_{i-1})^2 + 3h^2(x - x_{i-1}), & x \in [x_{i-1}, x_i],\\ h^3 - 3(x_{i+1} - x)^3 + 3h(x_{i+1} - x)^2 + 3h^2(x_{i+1} - x), & x \in [x_i, x_{i+1}],\\ (x_{i+2} - x)^3, & x \in [x_{i+1}, x_{i+2}],\\ 0, & \text{Otherwise.}\end{cases} \tag{8}$$

The B-spline $\{B_{-1}, B_0, ..., B_N, B_{N+1}\}$ establishes a basis over the assumed domain. The values of $B_i(x)$ at the different knots are given in Table 1.

Table 1. $B_i(x)$, $B_i{}'(x)$, and $B_i{}''(x)$ at the knots.

	x_{i+1}	x_i	x_{i-1}	Else
$B_i(x)$	$1/6$	$2/3$	$1/6$	0
$B_i{}'(x)$	$-1/2h$	0	$1/2h$	0
$B_i{}''(x)$	$1/h^2$	$-2/h^2$	$1/h^2$	0

The solution to Equation (4) is considered a linear combination of the CBS by joining the $N + 1$ number of control points with lines from $\mu_{-1}(t)$ to $\mu_{N+1}(t)$ and $\nu_{-1}(t)$ to $\nu_{N+1}(t)$ [34]. Our approach for the TFCBE with CBS is to seek as approximate solutions $U(x, t)$ and $V(x, t)$ in the following form [35]:

$$\begin{cases} U(x,t) = \sum_{i=-1}^{N+1} \mu_i(t) B_i(x),\\ V(x,t) = \sum_{i=-1}^{N+1} \nu_i(t) B_i(x),\end{cases} \tag{9}$$

where $\mu_i(t)$ and $\nu_i(t)$ are control points (or de Boor points), which depend on time and are to be computed at each time level for the approximate solution. $B_i(x)$ represents the CBS basis functions defined in (8). They are piecewise CBS functions with geometric properties like C^2 continuity, non-negativity, and partition of unity [36]. The approximations at each sub-interval $[x_i, x_{i-1}]$ contain $B_{i-1}(x), B_i(x), B_{i+1}(x)$ non-zero basis functions.

3. Discretization

The forward FDM is applied to discretize the Caputo time FD. Consider $t_k = k\tau$, where $k = 0, 1, ..., m$ and the step size is $\tau = \frac{T}{m}$. The Caputo time FD can be described as follows at point $t = t_k$:

$$\begin{aligned}\frac{\partial^\lambda U(x, t_{k+1})}{\partial t^\lambda} &= \frac{1}{\Gamma(1-\lambda)}\int_0^t \frac{\partial U(x,\sigma)}{\partial\sigma}\frac{d\kappa}{(t_{k+1}-\sigma)^\lambda}\\ &= \frac{1}{\Gamma(1-\lambda)}\sum_{p=0}^{k}\int_{t_p}^{t_{p+1}} \frac{\partial U(x,\sigma)}{\partial\sigma}\frac{d\kappa}{(t_{k+1}-\sigma)^\lambda}\\ &= \frac{1}{\Gamma(1-\lambda)}\sum_{p=0}^{k}\frac{U(x,t_{p+1}) - U(x,t_p)}{\tau}\int_{p\tau}^{(p+1)\tau}\frac{d\kappa}{(t_{k+1}-\kappa)^\lambda} + Z_\tau^{k+1}\\ &= \frac{1}{\Gamma(1-\lambda)}\sum_{p=0}^{k}\frac{U\Big(x,t_{1-p+k}\Big) - U\Big(x,t_{k-p}\Big)}{\tau}\int_{p\tau}^{(p+1)\tau}\frac{d\varpi}{\varpi^\lambda} + Z_\tau^{k+1}\\ &= \frac{1}{\Gamma(1-\lambda)}\sum_{p=0}^{k}\frac{U\Big(x,t_{1-p+k}\Big) - U\Big(x,t_{k-p}\Big)}{\tau^\lambda}\left[(p+1)^{-\lambda+1} - p^{-\lambda+1}\right] + Z_\tau^{k+1}.\end{aligned} \tag{10}$$

Equation (10) becomes

$$\frac{\partial^\lambda U(x,t_{\sigma+1})}{\partial t^\lambda} = \frac{1}{\Gamma(2-\lambda)} \sum_{p=0}^{k} Y_p \frac{U(x,t_{1-p+k}) - U(x,t_{k-p})}{\tau^\lambda} + Z_\tau^{k+1}, \tag{11}$$

where $Y_p = (p+1)^{-\lambda+1} - p^{-\lambda+1}$, and the truncation error is Z_τ^{k+1}, which is bounded, i.e., $|Z_\tau^{k+1}| \leq \theta \tau^{2-\lambda}$ [37]. The coefficients Y_p preserve the following properties [38]:

1. $Y_0 = 1$,
2. $Y_0 > Y_1 > Y_2 > \ldots > Y_p, Y_p \to 0 as p \to \infty$,
3. $Y_p > 0 for p = 0, 1, \ldots k$, and
4. $\sum_{p=0}^{k}(Y_p - Y_{p+1}) + Y_{k+1} = (1 - Y_1) + \sum_{p=1}^{k-1}(Y_p - Y_{p+1}) + Y_k = 1$.

4. Numerical Technique

The θ-weighted scheme [39,40] for the system (4) at the t_{k+1}^{th} time level is defined as

$$\begin{cases} (U_t)_j^k - \theta p_j^{k+1} = (1-\theta) p_j^k, \\ (V_t)_j^k - \theta q_j^{k+1} = (1-\theta) q_j^k, \end{cases} \tag{12}$$

where

$$\begin{cases} p_j^k = (U_{xx})_j^k - \gamma (UU_x)_j^k - \alpha_1 [(UV_x)_j^k + (U_x V)_j^k], \\ q_j^k = (V_{xx})_j^k - \gamma (VV_x)_j^k - \alpha_1 [(VU_x)_j^k + (V_x U)_j^k]. \end{cases}$$

It is noted that, when $\theta = 0$, the above scheme becomes explicit; when $\theta = 1$, it reduces to an implicit scheme; when $\theta = 1/2$, it reduces to a Crank–Nicolson scheme. Using (11) and the θ-weighted scheme (12) on the spatial derivative, Equation (4) then becomes

$$\begin{aligned} \frac{1}{\Gamma(2-\lambda)} \sum_{p=0}^{k} Y_p \frac{U_j^{k-p+1} - U_j^{k-p}}{\tau^\lambda} &- \theta[(U_{xx})_j^{k+1} - \gamma (UU_x)_j^{k+1} - \alpha_1 [(UV_x)_j^{k+1} + (U_x V)_j^{k+1}]] \\ &= (1-\theta)[(U_{xx})_j^k - \gamma (UU_x)_j^k - \alpha_1 [(UV_x)_j^k + (U_x V)_j^k]] + Q_1, \end{aligned} \tag{13}$$

$$\begin{aligned} \frac{1}{\Gamma(2-\lambda)} \sum_{p=0}^{k} Y_p \frac{V_j^{k-p+1} - V_j^{k-p}}{\tau^\lambda} &- \theta[(V_{xx})_j^{k+1} - \gamma (VV_x)_j^{k+1} - \alpha_2 [(VU_x)_j^{k+1} + (V_x U)_j^{k+1}]] \\ &= (1-\theta)[(V_{xx})_j^k - \gamma (VV_x)_j^k - \alpha_2 [(VU_x)_j^k + (V_x U)_j^k]] + Q_2, \end{aligned} \tag{14}$$

The quasi-linearization technique [41] at the $(k+1)^{th}$ stage is defined as

$$\begin{cases} (UU_x)_j^{k+1} = U_j^k (U_x)_j^{k+1} - (UU_x)_j^k + U_j^{k+1} (U_x)_j^k, \\ (VV_x)_j^{k+1} = V_j^k (V_x)_j^{k+1} - (VV_x)_j^k + V_j^{k+1} (V_x)_j^k, \\ (UV_x)_j^{k+1} = U_j^k (V_x)_j^{k+1} - (UV_x)_j^k + U_j^{k+1} (V_x)_j^k, \\ (VU_x)_j^{k+1} = V_j^k (U_x)_j^{k+1} - (VU_x)_j^k + V_j^{k+1} (U_x)_j^k. \end{cases} \tag{15}$$

Using the quasi-linearization technique (15), the non-linear terms in Equations (13) and (14) are linearized as

$$\begin{aligned} &\frac{1}{\Gamma(2-\lambda)} \sum_{p=0}^{k} Y_p \frac{U_j^{k-p+1} - U_j^{k-p}}{\tau^\lambda} - \theta[(U_{xx})_j^{k+1} - \gamma [U_j^k (U_x)_j^{k+1} - (UU_x)_j^k + U_j^{k+1} (U_x)_j^k] \\ &- \alpha_1 [U_j^k (V_x)_j^{k+1} - (UV_x)_j^k + U_j^{k+1} (V_x)_j^k + (U_x V)_j^{k+1}]] = (1-\theta)[(U_{xx})_j^k - \gamma (UU_x)_j^k - \alpha_1 \left((UV_x)_j^k + (U_x V)_j^k \right)] + Q_1, \end{aligned} \tag{16}$$

$$\begin{aligned} &\frac{1}{\Gamma(2-\lambda)} \sum_{p=0}^{k} Y_p \frac{V_j^{k-p+1} - V_j^{k-p}}{\tau^\lambda} - \theta[(V_{xx})_j^{k+1} - \gamma [U_j^k (V_x)_j^{k+1} - (UV_x)_j^k + U_j^{k+1} (V_x)_j^k] \\ &- \alpha_2 [V_j^k (U_x)_j^{k+1} - (VU_x)_j^k + V_j^{k+1} (U_x)_j^k + (V_x U)_j^{k+1}]] = (1-\theta)[(V_{xx})_j^k - \gamma (VV_x)_j^k - \alpha_2 \left((VU_x)_j^k + (V_x U)_j^k \right)] + Q_2, \end{aligned} \tag{17}$$

After some simplification of Equations (16) and (17), we obtain

$$\frac{1}{\Gamma(2-\lambda)}\sum_{p=0}^{k} Y_p \frac{U_j^{k-p+1}-U_j^{k-p}}{\tau^\lambda} - \theta(U_{xx})_j^{k+1} + \theta[\gamma(U_x)_j^k + \alpha_1(V_x)_j^k]U_j^{k+1} + \theta[\gamma U_j^k + \alpha_1 V_j^k](U_x)_j^{k+1}$$
$$+\alpha_1\theta[U_j^k(V_x)_j^{k+1} + (U_x)_j^k V_j^{k+1}] = (1-\theta)(U_{xx})_j^k - \gamma(1-2\theta)(UU_x)_j^k - \alpha_1(1-2\theta)(UV_x)_j^k - \alpha_1(1-2\theta)(U_xV)_j^k + Q_1, \quad (18)$$

$$\frac{1}{\Gamma(2-\lambda)}\sum_{p=0}^{k} Y_p \frac{V_j^{k-p+1}-V_j^{k-p}}{\tau^\lambda} - \theta(V_{xx})_j^{k+1} + \theta[\gamma(V_x)_j^k + \alpha_2(U_x)_j^k]V_j^{k+1} + \theta[\gamma V_j^k + \alpha_2 U_j^k](V_x)_j^{k+1}$$
$$+\alpha_2\theta[V_j^k(U_x)_j^{k+1} + (V_x)_j^k U_j^{k+1}] = (1-\theta)(V_{xx})_j^k - \gamma(1-2\theta)(VV_x)_j^k - \alpha_2(1-2\theta)(VU_x)_j^k - \alpha_2(1-2\theta)(V_xU)_j^k + Q_2, \quad (19)$$

The Crank–Nicolson approach is used because it is an unconditionally stable scheme and provides a more reasonable accuracy [42] than other finite difference schemes. Equations (18) and (19) provide the following relations:

$$\eta_1 Y_0 U_j^{k+1} - \tfrac{1}{2}(U_{xx})_j^{k+1} + \tfrac{\gamma_1}{2}U_j^{k+1} + \tfrac{\gamma_2}{2}(U_x)_j^{k+1} + \tfrac{\alpha_1}{2}[\gamma_3(V_x)_j^{k+1} + \gamma_4 V_j^{k+1}] =$$
$$\tfrac{1}{2}(U_{xx})_j^k - \eta_1\sum_{p=1}^{k} Y_p\left(U_j^{k-p+1} - U_j^{k-p}\right) + (Q_1)_j^{k+1}, \quad (20)$$

$$\eta_1 Y_0 V_j^{k+1} - \tfrac{1}{2}(V_{xx})_j^{k+1} + \tfrac{\gamma_1}{2}V_j^{k+1} + \tfrac{\gamma_2}{2}(V_x)_j^{k+1} + \tfrac{\alpha_2}{2}[\gamma_3(U_x)_j^{k+1} + \gamma_4 U_j^{k+1}] =$$
$$\tfrac{1}{2}(V_{xx})_j^k - \eta_1\sum_{p=1}^{k} Y_p\left(V_j^{k-p+1} - V_j^{k-p}\right) + (Q_2)_j^{k+1}, \quad (21)$$

where
$\eta_1 = \frac{1}{\tau^\lambda\Gamma(2-\lambda)}$,
$\gamma_1 = \gamma(U_x)_j^k + \alpha_1(V_x)_j^k = \gamma(V_x)_j^k + \alpha_2(U_x)_j^k$,
$\gamma_2 = \gamma U_j^k + \alpha_1 V_j^k = \gamma V_j^k + \alpha_2 U_j^k$, and
$\gamma_3 = U_j^k = V_j^k$ and $\gamma_3 = (U_x)_j^k = (V_x)_j^k$.
The boundary conditions are utilized to obtain a numerical solution to the proposed problem. Four extra linear equations are designed, as follows:

$$\begin{cases} (U)_0^{k+1} = \varphi_1(t_{k+1}), \\ (U)_n^{k+1} = \varphi_1(t_{k+1}), \\ (V)_0^{k+1} = \psi_1(t_{k+1}), and \\ (V)_n^{k+1} = \psi_1(t_{k+1}). \end{cases}$$

The initial conditions are listed below:

$$\begin{cases} (U_x)_m^{k+1} = \phi_1'(t_m), \quad m = 0, M \\ (U)_m^{k+1} = \phi_1(t_m), \quad m = 0,1,2,\ldots,M \\ (V_x)_m^{k+1} = \phi_2'(t_m), \quad m = 0, M \\ (V)_m^{k+1} = \phi_2(t_m), \quad m = 0,1,2,\ldots,M. \end{cases}$$

Equations (20) and (21) can be written in matrix form as follows:

$$AC^{k+1} = C^k B + D[\eta_1 Y_k \mu^0 + \sum_{p=0}^{k-1}(Y_p - Y_{p+1})\mu^{p-s} + \eta_1 Y_k \nu^0 + \sum_{p=0}^{k-1}(Y_p - Y_{p+1})\nu^{p-s}] + q^{k+1}. \quad (22)$$

where $C^{k+1} = [\mu_{-1}^{k+1}, \mu_0^{k+1}, \ldots \mu_{M+1}^{k+1}, \nu_{-1}^{k+1}, \nu_0^{k+1}, \ldots \nu_{M+1}^{k+1}]^T$ and $C^k = [\mu_{-1}^k, \mu_0^k, \ldots \mu_{M+1}^k, \nu_{-1}^k, \nu_0^k, \ldots \nu_{M+1}^k]^T$.
After some simplification, we obtain

$$AC^{k+1} = F. \quad (23)$$

where A represents the square matrix of order $(2M+6)\times(2M+6)$, and F is the column vector of order $(2M+6)$.

$$A=\begin{pmatrix} A_1 & \vdots & B_1\\ \cdots & \cdots & \cdots\\ B_1 & \vdots & A_1\end{pmatrix}, B=\begin{pmatrix} A_2 & \vdots & B_2\\ \cdots & \cdots & \cdots\\ B_2 & \vdots & A_2\end{pmatrix},$$

$$D=\begin{pmatrix} A_3 & \vdots & B_2\\ \cdots & \cdots & \cdots\\ B_2 & \vdots & A_3\end{pmatrix}, q^{k+1}=\begin{pmatrix} q_1\\ \cdots\\ q_2\end{pmatrix}.$$

where

$$A_1=\begin{pmatrix} \frac{1}{6} & \frac{4}{6} & \frac{1}{6} & & & & \\ p_1 & p_2 & p_1 & & & & \\ & p_1 & p_2 & p_1 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & p_1 & p_2 & p_1 & \\ & & & & p_1 & p_2 & p_1\\ & & & & \frac{1}{6} & \frac{4}{6} & \frac{1}{6}\end{pmatrix}, \quad B_1=\begin{pmatrix} 0 & 0 & 0 & & & & \\ p_3 & p_4 & p_3 & & & & \\ & p_3 & p_4 & p_3 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & p_3 & p_4 & p_3 & \\ & & & & p_3 & p_4 & p_3\\ & & & & 0 & 0 & 0\end{pmatrix},$$

$$A_2=\begin{pmatrix} 0 & 0 & 0 & & & & \\ p_5 & p_6 & p_5 & & & & \\ & p_5 & p_6 & p_5 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & p_5 & p_6 & p_5 & \\ & & & & p_5 & p_6 & p_5\\ & & & & 0 & 0 & 0\end{pmatrix}, \quad B_2=\begin{pmatrix} 0 & 0 & 0 & & & & \\ 0 & 0 & 0 & & & & \\ & 0 & 0 & 0 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & 0 & 0 & 0 & \\ & & & & 0 & 0 & 0\\ & & & & 0 & 0 & 0\end{pmatrix},$$

$$A_3=\begin{pmatrix} 0 & 0 & 0 & & & & \\ \frac{1}{6} & \frac{4}{6} & \frac{1}{6} & & & & \\ & \frac{1}{6} & \frac{4}{6} & \frac{1}{6} & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \frac{1}{6} & \frac{4}{6} & \frac{1}{6} & \\ & & & & \frac{1}{6} & \frac{4}{6} & \frac{1}{6}\\ & & & & 0 & 0 & 0\end{pmatrix}, \quad q_1=\begin{pmatrix} \varphi_1^{k+1}\\ (Q_1)_0^{k+1}\\ (Q_1)_1^{k+1}\\ \vdots\\ (Q_1)_{N-1}^{k+1}\\ (Q_1)_N^{k+1}\\ \varphi_2^{k+1}\end{pmatrix}, \quad q_2=\begin{pmatrix} \phi_1^{k+1}\\ (Q_2)_0^{k+1}\\ (Q_2)_1^{k+1}\\ \vdots\\ (Q_2)_{N-1}^{k+1}\\ (Q_2)_N^{k+1}\\ \phi_2^{k+1}\end{pmatrix},$$

where
$p_1=\frac{\eta_1 Y_0}{6}-\frac{1}{2h^2}+\frac{1}{12}\gamma_1-\frac{1}{4h}\gamma_2$,
$p_1=\frac{2\eta_1 Y_0}{3}-\frac{1}{h^2}+\frac{1}{3}\gamma_1$,
$p_3=\frac{\alpha}{2}\left(-\frac{\gamma_3}{2h}+\frac{\gamma_4}{6}\right)$,
$p_4=\frac{\alpha}{2}\left(\frac{\gamma_3}{2h}+\frac{2\gamma_4}{3}\right)$,
$p_5=\frac{1}{2h^2}+\frac{\eta_1 Y_0}{6}$, and
$p_6=-\frac{1}{h^2}+\frac{2\eta_1 Y_0}{3}$.

Algorithm

- Describe the model of TFCBEs.
- Use quasi-linearization technique to linearize the problem.
- Use the Caputo FD and the CBS functions to discretize the presented problem.
- Obtain the system of order $(2M+6)\times(2M+6)$.

- For each time step, solve the above system using Wolfram Mathematica 11.

5. Stability Analysis

A stability analysis is introduced to affirm that the scheme does not amplify errors. The numerical error and stability of the numerical scheme are strongly related. If problems committed during one computing time step do not result in additional errors as the computations proceed, then the finite difference scheme is stable. When errors in a scheme do not change as calculations are performed, it is said to be unconditionally stable [42]. The principle of stability is connected to computation technique errors that do not rise as the method progresses [43]. In this section, we will utilize the Fourier series scheme to assess the stability of the suggested methodology [44,45]. In order to perform Fourier stability analysis, a temporary freeze is applied to the non-linear terms. As a result, we have frozen U and V as constants ω_1 and ω_2, respectively [46]. Only linear equations are appropriate for stability analysis. The following equation is constructed by substituting an approximation of Equation (18):

$$\begin{aligned} \eta_1 Y_0 U_j^{k+1} - \frac{1}{2}\sum_{\iota=0}^{M} D_{j\iota}^2 U_\iota^{k+1} + \frac{1}{2}(\eta\omega_1+\alpha_1\omega_2)\sum_{\iota=0}^{M} D_{j\iota}^1 U_\iota^{k+1} + \frac{\alpha_1\omega_1}{2}\sum_{\iota=0}^{M} D_{j\iota}^1 V_\iota^{k+1} = -\frac{1}{2}(\eta\omega_1+\alpha_1\omega_2)\sum_{\iota=0}^{M} D_{j\iota}^1 U_\iota^{k} \\ -\frac{\alpha_1\omega_1}{2}\sum_{\iota=0}^{M} D_{j\iota}^1 V_\iota^{k} + \frac{1}{2}\sum_{\iota=0}^{M} D_{j\iota}^2 U_\iota^{k} + \eta_1 Y_0 U_j^k - \eta_1\sum_{p=1}^{k} Y_p[U_j^{k-s+1} - U_j^{k-s}] + Q_1. \end{aligned} \tag{24}$$

Let $\widetilde{U}_j^k$ and $\widetilde{V}_j^k$ be the estimated solution to (24), and errors μ_j^k and ν_j^k can be presented as

$$\begin{cases} \mu_j^k = U_j^k - \widetilde{U}_j^k, \\ \nu_j^k = V_j^k - \widetilde{V}_j^k, \end{cases} \quad k = 0,1,\ldots M, j = 0,1,\ldots N-1 \tag{25}$$

Corresponding vectors are as follows:

$$\begin{cases} \mu^k = [\mu_1^k, \mu_2^k, \ldots \mu_{N-1}^k]^T, \\ \nu^k = [\nu_1^k, \nu_2^k, \ldots \nu_{N-1}^k]^T. \end{cases} \tag{26}$$

We analyze the stability of the scheme given in (24), and the source term is considered to be zero. We obtain the round-off error equation as follows:

$$\begin{aligned} &\left[-\frac{1}{2h^2} - \frac{\zeta}{4h} + \frac{\eta_1 Y_0}{6}\right]\mu_{j-1}^{k+1} + \left[\frac{2\eta_1 Y_0}{3} + \frac{1}{h^2}\right]\mu_j^{k+1} + \left[-\frac{1}{2h^2} + \frac{\zeta}{4h} + \frac{\eta_1 Y_0}{6}\right]\mu_{j-1}^{k+1} - \frac{\alpha_1\omega_1}{4h}\nu_{j-1}^{k+1} \\ &+\frac{\alpha_1\omega_1}{4h}\nu_{j+1}^{k+1} = \left[\frac{1}{2h^2} + \frac{\zeta}{4h} + \frac{\eta_1 Y_0}{6}\right]\mu_{j-1}^{k} + \left[\frac{2\eta_1 Y_0}{3} - \frac{1}{h^2}\right]\mu_j^{k+1} + \left[\frac{1}{2h^2} - \frac{\zeta}{4h} + \frac{\eta_1 Y_0}{6}\right]\mu_{j+1}^{k+1} \\ &+\frac{\alpha_1\omega_1}{4h}\nu_{j-1}^{k} - \frac{\alpha_1\omega_1}{4h}\nu_{j+1}^{k} + \eta_1\sum_{p=1}^{k} Y_p\left[\frac{1}{6}\left(\mu_{j-1}^{k-p+1} - \mu_{j-1}^{k-p}\right) + \frac{2}{3}\left(\mu_j^{k-p+1} - \mu_j^{k-p}\right) + \frac{1}{6}\left(\mu_{j+1}^{k-p+1} - \mu_{j+1}^{k-p}\right)\right], \end{aligned} \tag{27}$$

where $\zeta = (\eta\omega_1 + \alpha_1\omega_2)$.

Grid functions are defined as follows:

$$\mu^k(x) = \begin{cases} \mu_j^k, & x_j - \frac{h}{2} < x \le x_j + \frac{h}{2}, \\ 0, & 0 < x \le \frac{h}{2} \text{ or } L - \frac{h}{2} < x \le L, \end{cases} \quad , \quad \nu^k(x) = \begin{cases} \nu_j^k, & x_j - \frac{h}{2} < x \le x_j + \frac{h}{2}, \\ 0, & 0 < x \le \frac{h}{2} \text{ or } L - \frac{h}{2} < x \le L, \end{cases}$$

Fourier expansion for $\mu^k(x)$ and $\nu^k(x)$ can be represented as

$$\begin{cases} \mu^k(x) = \sum_{p=-\infty}^{\infty} X_k(p) e^{i2\pi px/L}, \\ \nu^k(x) = \sum_{p=-\infty}^{\infty} Y_k(p) e^{i2\pi px/L}, \end{cases} \tag{28}$$

where $X_k(p) = \frac{1}{L}\int_0^L \mu^k(x) e^{i2\pi px/L} dx$ and $Y_k(p) = \frac{1}{L}\int_0^L \nu^k(x) e^{i2\pi px/L} dx$.

The Parseval equation is given as [47]:

$$\left\{ \int_0^L \parallel \mu^k(x) \parallel^2 dx = \sum_{p=-\infty}^{\infty} \parallel X_k(p) \parallel^2, \quad \int_0^L \parallel \nu^k(x) \parallel^2 dx = \sum_{p=-\infty}^{\infty} \parallel Y_k(p) \parallel^2, \right. \tag{29}$$

Applying the Parseval equality [48], which is

$$\begin{cases} \int_0^L \parallel \mu^k(x) \parallel^2 dx & = \sum_{j=1}^{M-1} h \parallel \mu_j^k \parallel^2, \\ \int_0^L \parallel \nu^k(x) \parallel^2 dx & = \sum_{j=1}^{M-1} h \parallel \nu_j^k \parallel^2, \end{cases}$$

we have

$$\begin{cases} \parallel \mu^k \parallel_2^2 & = \sum_{\ell=-\infty}^{\infty} \parallel X_k(\ell) \parallel^2, \\ \parallel \nu^k \parallel_2^2 & = \sum_{\ell=-\infty}^{\infty} \parallel Y_k(\ell) \parallel^2. \end{cases}$$

Now we suppose the solution in the form of Fourier series analysis, described as follows:

$$\begin{cases} \mu_j^k = X_k e^{i\psi_x jh}, \\ \nu_j^k = Y_k e^{i\psi_x jh}, \end{cases} \tag{30}$$

where $\psi_x = (2\pi\ell/L)$. Substituting the above relations into (24), we obtain

$$\begin{aligned} &\left[\tfrac{\eta_1 Y_0}{3}\cos(\psi_x h) - \tfrac{1}{h^2}\cos(\psi_x h) + \tfrac{2\eta_1 Y_0}{3} + \tfrac{1}{h^2}\right] X_{k+1} + i\left[\tfrac{\zeta}{2h}\sin(\psi_x h)\right] X_{k+1} + i\left[\tfrac{\alpha_1\omega_1}{2h}\sin(\psi_x h)\right] Y_{k+1} \\ &= \left[\tfrac{\eta_1 Y_0}{3}\cos(\psi_x h) + \tfrac{1}{h^2}\cos(\psi_x h) + \tfrac{2\eta_1 Y_0}{3} - \tfrac{1}{h^2}\right] X_k - i\left[\tfrac{\zeta}{2h}\sin(\psi_x h)\right] X_k - i\left[\tfrac{\alpha_1\omega_1}{2h}\sin(\psi_x h)\right] Y_k \\ &\qquad + \eta_1 \sum_{p=1}^{k} \left[\left(X_{k-p+1} - X_{k-p}\right)\tfrac{1}{3}\cos(\psi_x h) + \tfrac{2}{3}\right]. \end{aligned} \tag{31}$$

Set

$$\begin{aligned} A &= \frac{\eta_1 Y_0}{3}\cos(\psi_x h) - \frac{1}{h^2}\cos(\psi_x h) + \frac{2\eta_1 Y_0}{3} + \frac{1}{h^2}, \\ B &= \frac{\zeta}{2h}\sin(\psi_x h), \\ C &= \frac{\alpha_1\omega_1}{2h}\sin(\psi_x h), \\ D &= \cos(\psi_x h). \end{aligned}$$

We have

$$AX_{k+1} + i(BX_{k+1} + CY_{k+1}) = -(A+iB)X_k - iCY_k + \frac{\eta_1}{3}\sum_{p=1}^{k}\left[\left(X_{k-p+1} - X_{k-p}\right)D + 2\right]$$

$$(A+iB)X_{k+1} + iCY_{k+1} = -(A+iB)X_k - iCY_k + \frac{\eta_1}{3}\sum_{p=1}^{k}\left[\left(X_{k-p+1} - X_{k-p}\right)D + 2\right]$$

$$(A+iB)X_{k+1} = -(A+iB)X_k - iC(Y_{k+1} + Y_k) + \frac{\eta_1}{3}\sum_{p=1}^{k}\left[\left(X_{k-p+1} - X_{k-p}\right)D + 2\right] \tag{32}$$

Then

$$|X_{k+1}| \leq | X_k | + \frac{| iC |}{| A+iB |}(| Y_{k+1} | + | Y_k |) + \frac{1}{| A+iB |}\frac{\eta_1}{3}\sum_{p=1}^{k}\left[\left(X_{k-p+1} - X_{k-p}\right)D + 2\right]. \tag{33}$$

Theorem 1. *Let X_k be the solution to (32). Then E_k is the positive constant, and we have*

$$| X_k | \leq E_k | X_0 |, k = 1, 2, 3, ..., N-1. \tag{34}$$

Proof. Using mathematical induction, let $k = 1$ in Equation (33) yield

$$|X_1| \leq | X_0 | + \frac{| iC |}{| A + iB |}(| Y_1 | + | Y_0 |). \tag{35}$$

Using the convergence of the series on the right-hand side of Equation (33),

$$| Y_k | \leq P_1 | X_0 |, k = 0, 1, 2, ..., N-1,$$

where P_1 is a positive constant. Equation (35) becomes

$$| X_1 | \leq | X_0 | + P_1(| X_0 |) \leq (1 + P_1) | X_0 | \leq E_1 | X_0 |,$$

where $E_1 = 1 + P_1$. Now, suppose that

$$| X_k | \leq E_k | X_0 |, k = 1, 2, ..., N-2. \tag{36}$$

Using Equation (36), Equation (33) yields

$$| X_{k+1} | \leq E_k | X_0 | + GE_1 | X_0 | + H \sum_{p=1}^{k} [(X_0 - X_0)D + 2] \leq E_{k+1} | X_0 |, \tag{37}$$

where $G = \frac{|iC|}{|A+iB|}$, and $H \frac{|1|}{|A+iB|}$. □

Remark 1. *Similar to the above, positive constants I_k are such that*

$$| Y_k | \leq I_k | Y_0 |, k = 1, 2, ..., N-1. \tag{38}$$

Theorem 2. *The FD scheme (20) and (21) are unconditionally stable for $\lambda \epsilon (0, 1)$.*

Proof. According to Theorem 1 and Remark 1, we obtain

$$\begin{cases} \| \mu^k \|_2^2 & = \sum_{\ell=-\infty}^{\infty} \| X_k(\ell) \|^2 \leq \sum_{\ell=-\infty}^{\infty} E_k^2 \| X_0(\ell) \|^2 = E_k^2 \| \mu^0 \|_2^2, \\ \| \nu^k \|_2^2 & = \sum_{\ell=-\infty}^{\infty} \| Y_k(\ell) \|^2 \leq \sum_{\ell=-\infty}^{\infty} I_k^2 \| Y_0(\ell) \|^2 = I_k^2 \| \nu^0 \|_2^2, \end{cases} \tag{39}$$

$$\begin{aligned} \| U^k - \widetilde{u}^k \|_2^2 &\leq E_k \| U^0 - \widetilde{u}^0 \|_2^2, \\ \| V^k - \widetilde{v}^k \|_2^2 &\leq I_k \| V^0 - \widetilde{v}^0 \|_2^2, \end{aligned} \tag{40}$$

Hence, the proposed scheme is unconditionally stable. □

6. Numerical Results

In this segment, we describe two examples to illustrate the performance of the scheme (18) and (19). Results are obtained by using the cubic B-spline. Absolute errors are calculated using the $\|L\|_\infty$ and $\|L\|_2$ errors, i.e.,

$$\|L\|_\infty = \|U(x_r, t) - \widetilde{u}(x_r, t)\|_\infty = \max_{0 \leq r \leq N} | U(x_r, t) - \widetilde{u}(x_r, t) |,$$

and

$$\|L\|_2 = \|U(x_r, t) - \widetilde{u}(x_r, t)\|_2 = \left(h \sum_{i=0}^{N} | U(x_r, t) - \widetilde{u}(x_r, t) |^2 \right)^{\frac{1}{2}}.$$

where $\widetilde{u}(x_r, t)$ is an approximate solution.

Example 1. *Consider the TFCBE in the Caputo sense, when $\gamma = 1000$ and $\alpha_1 = \alpha_2 = 0.001$,*

$$\begin{cases} \frac{\partial^\lambda U}{\partial t^\lambda} - \frac{\partial^2 U}{\partial x^2} + \gamma U \frac{\partial U}{\partial x} + \alpha_1 \frac{\partial (UV)}{\partial x} = Q_1 \\ \frac{\partial^\lambda V}{\partial t^\lambda} - \frac{\partial^2 V}{\partial x^2} + \gamma V \frac{\partial V}{\partial x} + \alpha_2 \frac{\partial (UV)}{\partial x} = Q_2 \end{cases} a \le x \le b, 0 \le t \le T, 0 < \lambda < 1 \tag{41}$$

where BCs and ICs are $\begin{cases} u(0,t) = u(1,t) = 0 \\ v(0,t) = v(1,t) = 0 \end{cases}$ *and* $\begin{cases} u(x,0) = 0 \\ v(x,0) = 0 \end{cases}$*, respectively.*

The exact solutions are $U(x,t) = V(x,t) = x^2(x-1)^2 t^2$ [48]. In Table 2, the computational results exhibit the value of $U(x,t)$ and $V(x,t)$ at different stages of x, for Example 1. In Tables 3 and 4, the $\|L\|_\infty$ of $U(x,t)$ are estimated on $\lambda = 0.1, 0.3$ and 0.5 for different values of N and provide an analysis of the convergence order between stages. In Table 5, $\|L\|_\infty$ and an order of convergence assessment are conducted for $U(x,t)$ at different values of τ when $\lambda = 0.1, 0.005$.

Table 2. Absolute error of $U(x,t)$ and $V(x,t)$ at $\lambda = 0.3$ for Example 1, when $\Delta t = \frac{1}{1000}$, and N = 100.

		For $U(x,t)$		For $V(x,t)$	
x	**Exact Solution**	**Approximated Solution**	**Error**	**Approximated Solution**	**Error**
0.1	0.0081	0.0080992	7.95989×10^{-7}	0.0080992	7.95986×10^{-7}
0.2	0.0256	0.0255991	8.90499×10^{-7}	0.0255991	8.90497×10^{-7}
0.3	0.0441	0.0440991	9.17458×10^{-7}	0.0440991	9.17457×10^{-7}
0.4	0.0576	0.0575990	1.01942×10^{-6}	0.0575990	1.01942×10^{-6}
0.5	0.0625	0.0624987	1.27097×10^{-6}	0.0624987	1.27097×10^{-6}
0.6	0.0576	0.0575982	1.82313×10^{-6}	0.0575982	1.82313×10^{-6}
0.7	0.0441	0.0440967	3.31485×10^{-6}	0.0440967	3.31485×10^{-6}
0.8	0.0256	0.0255910	9.02991×10^{-6}	0.0255910	9.02991×10^{-6}
0.9	0.0081	0.0080862	1.38116×10^{-5}	0.0080862	1.38116×10^{-5}

Table 3. Absolute error and order of convergence of $U(x,t)$ when $\lambda = 0.1, 0.3, 0.5$ and $\Delta t = 0.001$ for Example 1.

h	$\|L\|_\infty$	Order	$\|L\|_\infty$	Order	$\|L\|_\infty$	Order
	$\lambda = 0.1$		$\lambda = 0.3$		$\lambda = 0.5$	
1/20	3.39429×10^{-4}	···	3.39353×10^{-4}	···	3.39557×10^{-4}	···
1/40	8.88467×10^{-5}	1.93372	8.88382×10^{-5}	1.93353	8.88819×10^{-5}	1.93369
1/80	2.22587×10^{-5}	1.99695	2.22520×10^{-5}	1.99725	2.22364×10^{-5}	1.99897
1/160	5.59571×10^{-6}	1.99198	5.58911×10^{-6}	1.99324	5.55831×10^{-6}	2.00021
1/320	1.41227×10^{-6}	1.98632	1.40599×10^{-6}	1.98461	1.37167×10^{-6}	2.01871

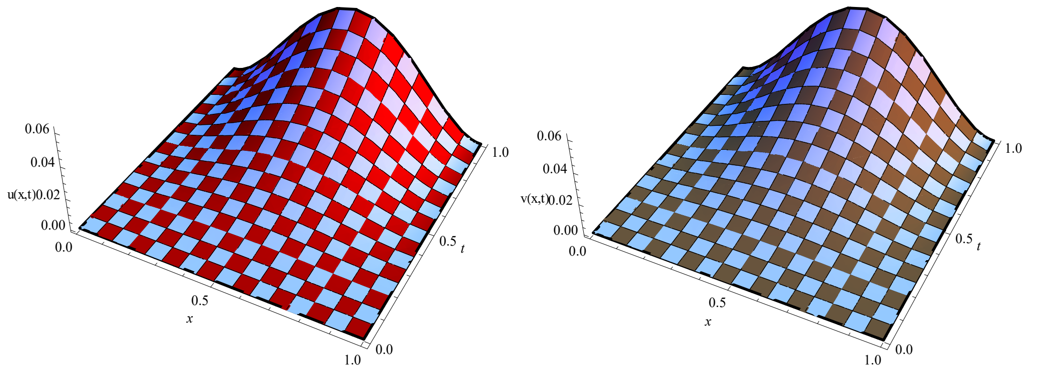
Table 4. Absolute error and order of convergence of $U(x,t)$ when $\lambda = 0.7, 0.8, 0.9$ and $\Delta t = 0.001$ for Example 1.

h	$\|L\|_\infty$	Order	$\|L\|_\infty$	Order	$\|L\|_\infty$	Order
	$\lambda = 0.7$		$\lambda = 0.8$		$\lambda = 0.9$	
1/20	3.39995×10^{-4}	···	3.40177×10^{-4}	···	3.40102×10^{-4}	···
1/40	8.88670×10^{-5}	1.93579	8.86860×10^{-5}	1.93951	8.81345×10^{-5}	1.94819
1/80	2.20861×10^{-5}	2.00851	2.18047×10^{-5}	2.02407	2.11294×10^{-5}	2.06046
1/160	5.37405×10^{-6}	2.03906	5.06770×10^{-6}	2.10524	4.36195×10^{-6}	2.27621
1/320	1.17911×10^{-6}	2.18831	8.66987×10^{-7}	2.54725	1.63195×10^{-6}	1.88726

Table 5. Absolute error and order of convergence of $U(x,t)$ when $\lambda = 0.1, 0.05$ and $N = 512$ for Example 1.

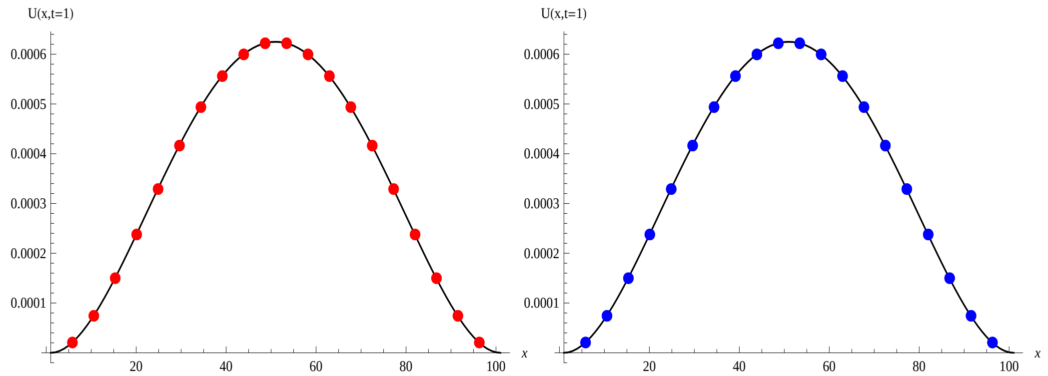
τ	$\|L\|_\infty$	Order	$\|L\|_\infty$	Order
	$\lambda = 0.1$		$\lambda = 0.05$	
1/16	4.41715×10^{-4}	$\cdots$	4.41624×10^{-4}	$\cdots$
1/32	1.14146×10^{-4}	1.95224	1.14119×10^{-4}	1.95228
1/64	2.89674×10^{-5}	1.97838	2.89596×10^{-5}	1.97843
1/128	7.26223×10^{-6}	1.99595	7.26003×10^{-6}	1.99599

Figure 1 shows the exact 3D solution, while Figure 2 represents the comparison of 2D exact and approximate values U and V using $N = 100$, $\Delta t = 0.001$, and $\lambda = 0.1$. The convergence of the exact and approximate solutions is illustrated in Figure 3 at different time stages for Example 1. In Figure 4, the graphs display the function error for different values of N and λ.



(**a**) (**b**)

Figure 1. Exact 3D solutions for Example 1. (**a**) For U; (**b**) For V.



(**a**) (**b**)

Figure 2. For Example 1, Comparison between 2D Exact and Approximate solution. (**a**) For U, (**b**) For V.

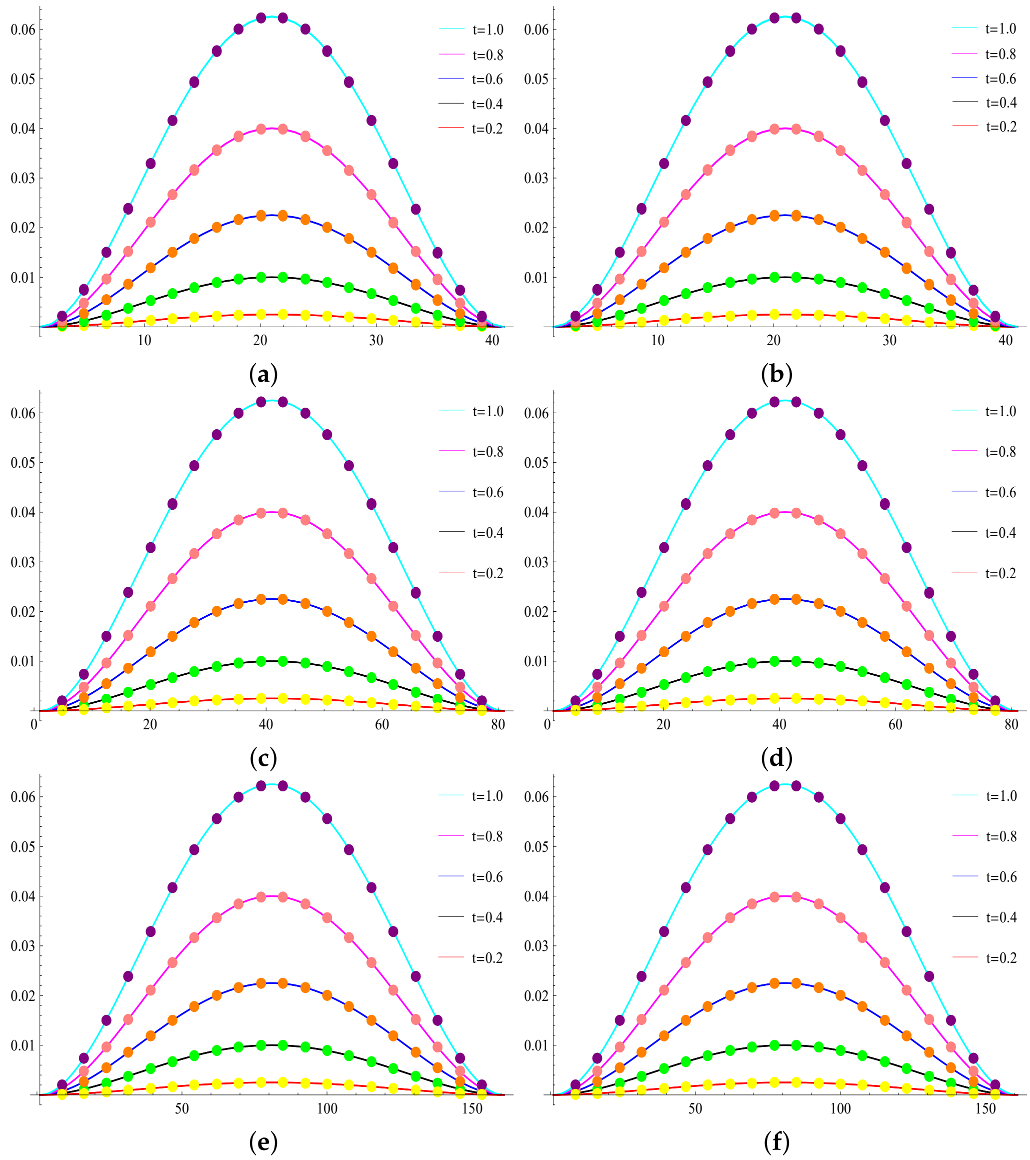


Figure 3. For Example 1, Exact and Approximate solutions at different time stages. (**a**) $N = 40$, $\Delta t = 1/32$ and $\lambda = 0.1$ of U, (**b**) $N = 40$, $\Delta t = 1/32$ and $\lambda = 0.1$ of V, (**c**) $N = 80$, $\Delta t = 1/64$ and $\lambda = 0.5$ of U, (**d**) $N = 80$, $\Delta t = 1/64$ and $\lambda = 0.5$ of V, (**e**) $N = 160$, $\Delta t = 1/132$ and $\lambda = 0.9$ of U, (**f**) $N = 160$, $\Delta t = 1/132$ and $\lambda = 0.9$ of V.

In Figure 4, graphs show the error function (EF) for different value of N and λ.

Example 2. *Consider the TFCBE in the Caputo sense, when $\gamma = 1000$ and $\alpha_1 = \alpha_2 = 0.001$,*

$$\begin{cases} \frac{\partial^\lambda U}{\partial t^\lambda} - \frac{\partial^2 U}{\partial x^2} + \gamma U \frac{\partial U}{\partial x} + \alpha_1 \frac{\partial (UV)}{\partial x} = Q_1 \\ \frac{\partial^\lambda V}{\partial t^\lambda} - \frac{\partial^2 V}{\partial x^2} + \gamma V \frac{\partial V}{\partial x} + \alpha_2 \frac{\partial (UV)}{\partial x} = Q_2 \end{cases} \quad a \le x \le b, 0 \le t \le T, 0 < \lambda < 1 \tag{42}$$

where BCs and ICs are $\begin{cases} u(0,t) = v(o,t) = t^2 \\ u(1,t) = v(1,t) = t^2 e \end{cases}$ *and* $\begin{cases} u(x,0) = 0 \\ v(x,0) = 0 \end{cases}$*, respectively.*

The exact solutions are $U(x,t) = V(x,t) = t^2 e^x$. For Example 2, Table 6 displays the computational results showing the values of the function at various stages, while Table 7 represents the behaviour of the error norm for different values of k at $\lambda = 0.1$ and 0.05.

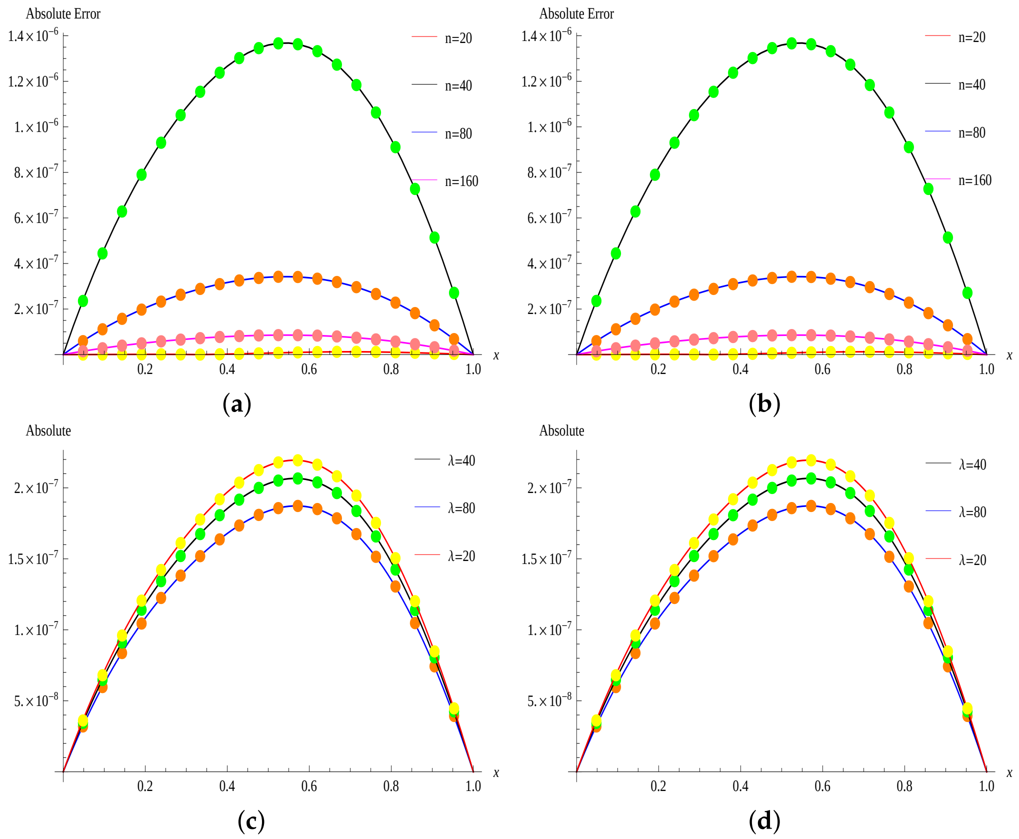


Figure 4. Error graph at different stages for Example 1. (**a**) EF for U, when $\lambda = 0.1$ and $\Delta t = 1/1000$. (**b**) EF for V, when $\lambda = 0.1$ and $\Delta t = 1/1000$. (**c**) EF for U, when $N = 100$ and $\Delta t = 1/100$. (**d**) EF for V, when $N = 100$ and $\Delta t = 1/100$.

Table 6. Absolute error of $U(x,t)$ and $V(x,t)$, when $\lambda = 0.1$, $\Delta t = 0.001$, and $N = 100$, for Example 2.

		For $U(x,t)$		For $V(x,t)$	
x	Exact Solution	Approximated Solution	Error	Approximated Solution	Error
0.1	1.10517	1.10517	2.99571×10^{-7}	1.10517	2.99571×10^{-7}
0.2	1.22140	1.22140	6.01967×10^{-7}	1.22140	6.01967×10^{-7}
0.3	1.34986	1.34986	9.10470×10^{-7}	1.34986	9.10470×10^{-7}
0.4	1.49182	1.49183	1.22815×10^{-6}	1.49183	1.22815×10^{-6}
0.5	1.64872	1.64872	1.55804×10^{-6}	1.64872	1.55804×10^{-6}
0.6	1.82212	1.82212	1.90185×10^{-6}	1.82212	1.90185×10^{-6}
0.7	2.01375	2.01375	2.25080×10^{-6}	2.01375	2.25080×10^{-6}
0.8	2.22554	2.22554	2.53167×10^{-6}	2.22554	2.53167×10^{-6}
0.9	2.71828	2.71828	2.35147×10^{-6}	2.71828	2.35147×10^{-5}

Table 7. Absolute error and order of convergence of $U(x,t)$ when $\lambda = 0.1, 0.05$ and $N = 512$ for Example 2.

τ	$\|L\|_\infty$	Order	$\|L\|_\infty$	Order
	$\lambda = 0.1$		$\lambda = 0.05$	
1/64	1.63599×10^{-3}	$\cdots$	1.63597×10^{-3}	$\cdots$
1/128	4.10724×10^{-4}	1.99392	4.10713×10^{-4}	2.01259
1/256	1.01791×10^{-4}	2.01256	1.01786×10^{-4}	2.01259
1/512	2.42205×10^{-5}	2.07131	2.42145×10^{-5}	2.07159

Figure 5 shows the exact 3D solution, while Figure 6 represents the comparison of the exact and approximate 2D solutions for U and V using $N = 100$, $\Delta t = 0.001$, and $\lambda = 0.1$. For Example 2, Figure 7 illustrates the convergence of the exact and approximate solution at different stages. In Figure 8, the graphs display the function error for different values of N and λ.

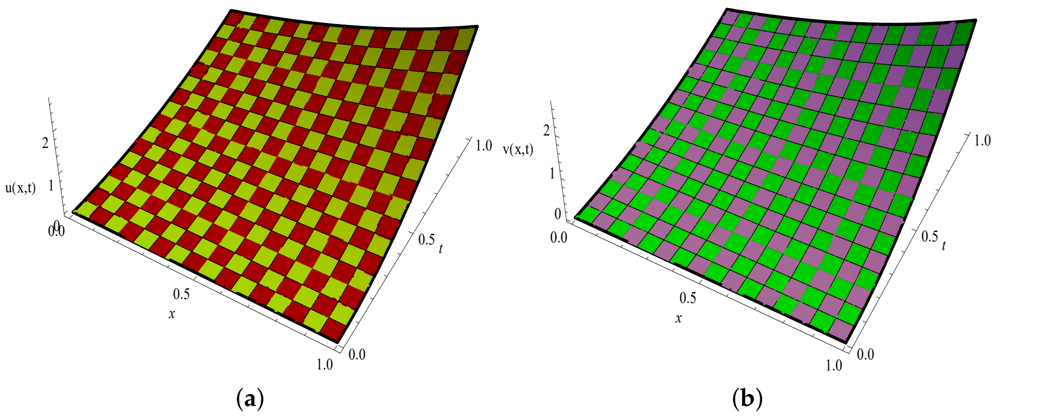


Figure 5. Exact 3D solutions for Example 2. (**a**) For U; (**b**) For V.

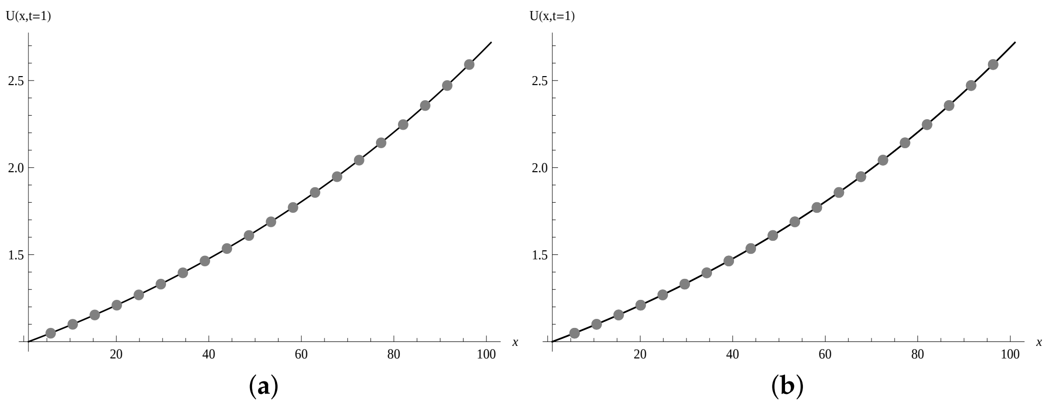


Figure 6. For Example 2 Comparison between Exact and Approximate 2D solution. (**a**) For U; (**b**) For V.

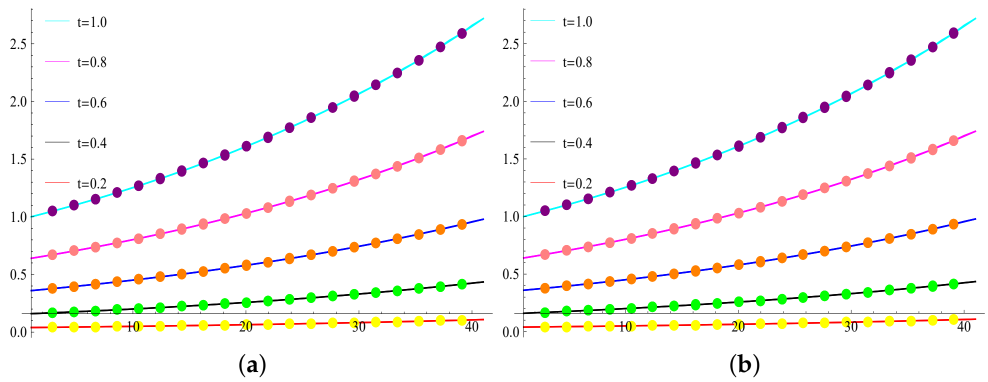


Figure 7. *Cont.*

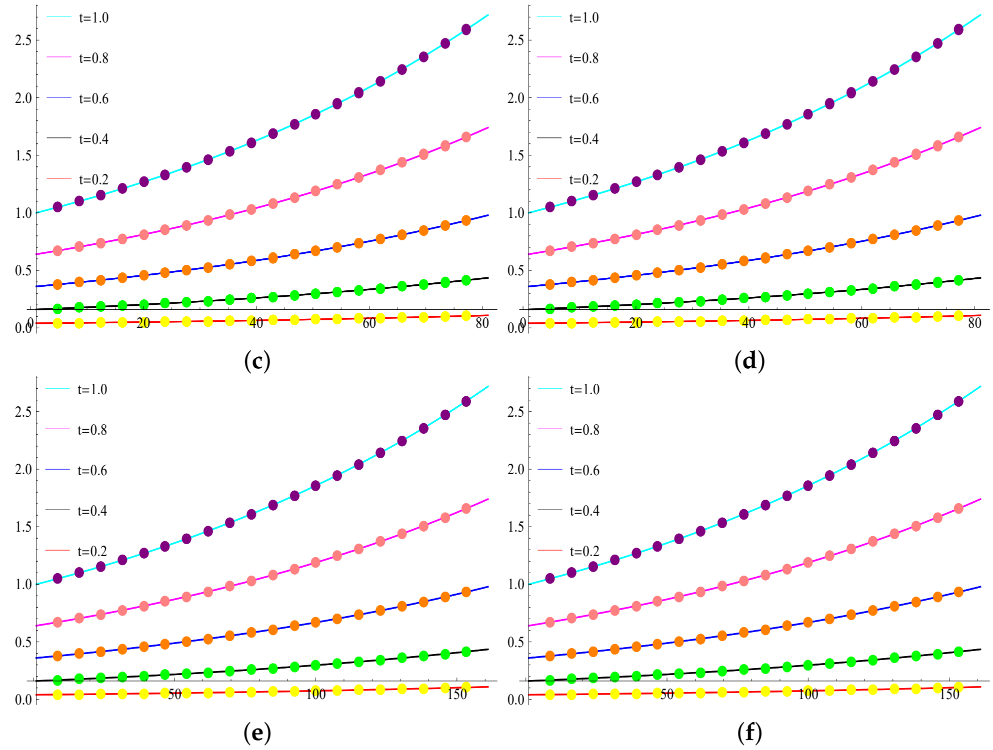


Figure 7. For Example 2, exact and approximate solutions at different time stages. (**a**) $N = 40$, $\Delta t = 1/32$, and $\lambda = 0.1$ of U. (**b**) $N = 40$, $\Delta t = 1/32$, and $\lambda = 0.1$ of V. (**c**) $N = 80$, $\Delta t = 1/64$, and $\lambda = 0.9$ of U. (**d**) $N = 80$, $\Delta t = 1/64$, and $\lambda = 0.9$ of V. (**e**) $N = 160$, $\Delta t = 1/132$, and $\lambda = 0.5$ of U. (**f**) $N = 160$, $\Delta t = 1/132$, and $\lambda = 0.5$ of V.

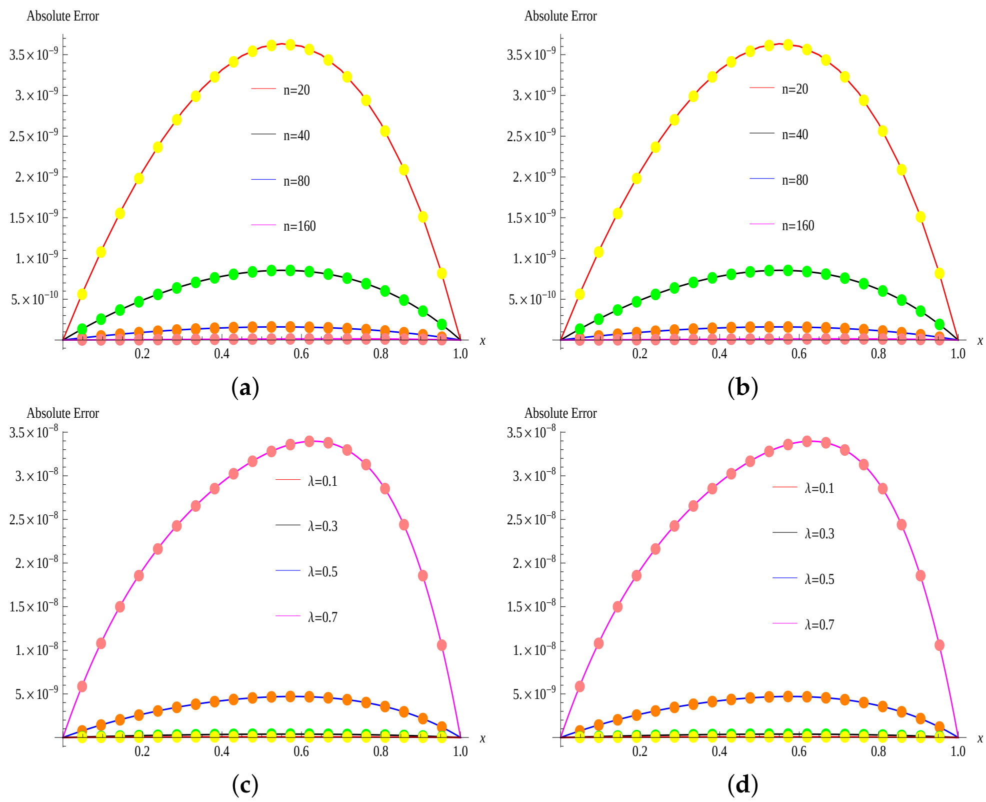


Figure 8. Error graph of function at different stages for Example 2. (**a**) EF for U, when $\Delta t = 1/512$ and $\lambda = 0.1$. (**b**) EF for V, when $\Delta t = 1/512$ and $\lambda = 0.1$. (**c**) EF for U, when $N = 100$ and $\Delta t = 1/100$. (**d**) EF for V, when $N = 100$ and $\Delta t = 1/100$.

7. Conclusion Remarks

In the conducted research, we have developed the efficient approximate method for TFCBEs. It is used to study turbulent fluids, suspensions, and the propagation of shallow water waves. Here, we obtain a approximate solution for TFCBEs using the CBS and Crank–Nicolson method. The time FD has been discretized using Caputo's formula. The numerical algorithm shows that the system is unconditionally stable. The numerical order of convergence has also been determined. Two numerical test problems have been considered in order to evaluate the effectiveness of the delivered method. The results presented in the tables and graphs demonstrate the applicability and accuracy of the presented technique. Approximated values have been compared with exact values, and their errors have been determined. The numerical results were calculated using Mathematica 12.3, and our method yielded acceptable outcomes.

Author Contributions: Conceptualization, U.G., M.A. and T.A.; Funding acquisition, E.K.E.-S., M.A.E.A. and N.F.A.; Investigation, U.G., M.A. and T.A.; Methodology, U.G., M.A. and T.A.; Project administration, E.K.E.-S., M.A.E.A. and N.F.A.; Resources, M.A.; Software, U.G., M.A., M.A.E.A. and N.F.A.; Supervision, M.A.; Validation, U.G., M.A., T.A., E.K.E.-S., M.A.E.A. and N.F.A.; Writing—original draft, U.G., M.A., T.A., E.K.E.-S., M.A.E.A. and N.F.A.; Writing—review & editing, U.G., E.K.E.-S., M.A.E.A. and N.F.A. All authors have read and agreed to the published version of the manuscript.

Funding: The Deputyship for Research & Innovation in the Ministry of Education in Saudi Arabia funded this research work through project number 445-9-492.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors extend their appreciation to the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia for funding this research work (project number 445-9-492). The authors are also grateful to the anonymous referees for their valuable suggestions, which significantly improved this manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Burgers, J.M. A mathematical model illustrating the theory of turbulence. *Adv. Appl. Mechanics.* **1948**, *1*, 171–199.
2. Cole, J.D. On a quasi-linear parabolic equation occurring in aerodynamics. *Q. Appl. Math.* **1951**, *9*, 225–236. [CrossRef]
3. Esipov, S.E. Coupled Burgers equations: A model of polydispersive sedimentation. *Phys. Rev. E* **1995**, *52*, 3711. [CrossRef] [PubMed]
4. Nee, J.; Duan, J. Limit set of trajectories of the coupled viscous Burgers equations. *Appl. Math. Lett.* **1998**, *11*, 57–61. [CrossRef]
5. Logan, J.D. *An Introduction to Nonlinear Partial Differential Equations*; John Wiley & Sons: Hoboken, NJ, USA, 2008; Volume 89.
6. Caputo, M.; Fabrizio, M. A new definition of fractional derivative without singular kernel. *Prog. Fract. Differ. Appl.* **2015**, *1*, 113.
7. Li, C.; Zeng, F. *Numerical Methods for Fractional Calculus*; CRC Press: New York, NY, USA, 2015.
8. Gorenflo, R.; Mainardi, F. *Fractional Calculus: Integral and Differential Equations of Fractional Order*; Springer: Vienna, Austria, 1997; pp. 223–276.
9. Chen, Y.; An, L.H. Numerical solutions of coupled Burgers equations with time and space-fractional derivatives. *Appl. Math. Comput.* **2008**, *200*, 87–95. [CrossRef]
10. Khan, N.A.; Ara, H.; Mahmood, A. Numerical solutions of time-fractional Burgers equations. *Int. J. Numer. Methods Heat Fluid Flow* **2012**, *22*, 175–193.
11. Prakash, A.; Kumar, M.; Sharma, K.K. Numerical method for solving fractional coupled Burgers equations. *Int. J. Numer. Methods Heat Fluid Flow* **2015**, *260*, 314–320. [CrossRef]
12. Singh, J.; Kumar, D.; Swroop, R. Numerical solution of time- and space-fractional coupled Burgers' equations via homotopy algorithm. *Alex. Eng. J.* **2016**, *55*, 1753–1763. [CrossRef]
13. Aminikhah, H.; Malekzadeh, N. An efficient method for systems of variable coefficient Coupled Burgers' equation with time-fractional derivative. *Sci. World J.* **2013**, *2013*, 687695. [CrossRef]
14. Ahmed, H.F.; Baghat, M.S.M.; Zaki, M. Analytical approaches to space and time fractional coupled Burgers' equations. *Pramana* **2019**, *92*, 38. [CrossRef]
15. Albouhimad, B.; Adibi, H. On a hybrid spectral exponential Chebyshev method for time-fractional coupled Burgers equations on a semi-infinite domain. *Adv. Differ. Equ.* **2017**, *2017*, 85. [CrossRef]

16. Sulaiman, T.A.; Yavuz, M.; Bulut, H.; Baskonus, H.M. Investigation of the fractional coupled viscous Burgers; equation involving Mittag-Leffler kernel. *Stat. Mech. Its Appl.* **2019**, *527*, 121–126. [CrossRef]
17. Liu, J.; Hou, G. Numerical solutions of the space and time fractional coupled Burgers equations by generalized differential transform method. *Appl. Math. Comput.* **2011**, *217*, 7001–7008. [CrossRef]
18. Ozdemir, N.; Secer, A.; Bayram, M. The gegenbauer wavelets-based computational methods for the coupled system of Burgers equations with time fractional derivative. *Mathematics* **2019**, *7*, 486. [CrossRef]
19. Abazari, R.; Borhanifar, A. Numerical study of the solution of the burgers and coupled burgers equations by a differential transformation method. *Comput. Appl.* **2010**, *59*, 2711–2722. [CrossRef]
20. Mittal, R.; Arora, G. Numerical solution of the coupled viscous burgers equation. *Commun. Nonlinear Sci. Numer. Simul.* **2011**, *16*, 1304–1313. [CrossRef]
21. Shukla, H.; Tamsir, M.; Srivastava, V.K.; Kumar, J. Numerical solution of two dimensional coupled viscous burger equation using modified cubic b-spline differential quadrature method. *AIP Adv.* **2014**, *4*, 117134. [CrossRef]
22. Kumar, A.; Arora, R. Solutions of the coupled system of burgers equations and coupled klein-gordon equation by rdt method. *Int. J. Adv. Appl. Math. Mech.* **2013**, *1*, 103–115.
23. Srivastava, V.K.; Tamsir, M.; Awasthi, M.K.; Singh, S. One dimensional coupled burgers equation and its numerical solution by an implicit logarithmic finite difference method. *AIP Adv.* **2014**, *4*, 037119. [CrossRef]
24. Mittal, R.; Tripathi, A. A collocation method for numerical solutions of coupled burgers equations. *Int. J. Comput. Methods Eng. Sci. Mech.* **2014**, *15*, 457–471. [CrossRef]
25. Sarboland, M.; Aminataei, A. An efficient numerical scheme for coupled nonlinear burgers equations. *Appl. Math. Inf. Sci.* **2015**, *9*, 245–255. [CrossRef]
26. Salih, H.M.; Tawfiq, L.N.M.; Yahya, Z.R. Numerical solution of the coupled viscous burgers equation via cubic trigonometric b-spline approach. *Math Stat.* **2016**, *2*, MSOA-2-011.
27. He, Y.; Tang, X. Numerical simulations of a family of the coupled viscous burgers equation using the lattice Boltzmann method3. *J. Stat. Mech. Theory Exp.* **2016**, *2016*, 023208. [CrossRef]
28. Chuathong, N.; Kaennakham, S. Numerical solution to coupled burgers equations by gaussian based hermite collocation scheme. *J. Appl. Math.* **2018**, *2018*, 3416860. [CrossRef]
29. Jima, M.; Shiferaw, A.; Tsegaye, A. Numerical solution of the coupled viscous burgers equation using differential quadrature method based on fourier expansion basis. *Appl. Math.* **2018**, *9*, 821. [CrossRef]
30. Oruc, O.; Bulut, F.; Esen, A. Chebyshev wavelet method for numerical solutions of coupled burgers equation. *Hacet. J. Math. Stat.* **2019**, *48*, 1–16.
31. Wu, G.C.; Zeng, D.Q.; Baleanu, D. Fractional impulsive differential equations: Exact solutions, integral equations and short memory case. *Fract. Calc. Appl. Anal.* **2019**, *22*, 180–192. [CrossRef]
32. Dubey, V.P.; Singh, J.; Alshehri, A.M.; Dubey, S.; Kumar, D. Analysis of local fractional coupled Helmholtz and coupled Burgers' equations in fractal media. *AIMS Math.* **2022**, *7*, 8080–8111. [CrossRef]
33. Wang, K.J. New exact solutions of the local fractional modified equal width-Burgers equation on the Cantor sets. *FRACTALS (Fractals)* **2023**, *31*, 1–9. [CrossRef]
34. Nazir, T.; Abbas, M.; Iqbal, M.K. New cubic B-spline approximation technique for numerical solutions of coupled viscous Burgers equations. *Eng. Comput.* **2021**, *38*, 83–106. [CrossRef]
35. De Boor, C.; De Boor, C. *A Practical Guide to Splines*; Springer: New York, NY, USA, 1978; Volume 27, p. 325.
36. Koch, P.E.; Lyche, T.; Neamtu, M.; Schumaker, L.L. Control curves and knot insertion for trigonometric splines. *Adv. Comput. Math.* **1995**, *3*, 405–424. [CrossRef]
37. Lin, Y.; Xu, C. Finite difference/spectral approximations for the timefractional diffusion equation. *J. Comput. Phys.* **2007**, *2225*, 1533–1552. [CrossRef]
38. Sayevand, K.; Yazdani, A.; Arjang, F. Cubic B-spline collocation method and its application for anomalous fractional diffusion equations in transport dynamic systems. *J. Vib. Control* **2016**, *22*, 2173–2186. [CrossRef]
39. Ru, J.G.Y. B-Spline for Initial and Boundary Value Problems. Ph.D. Thesis, Universiti Sains Malaysia, Penang, Malaysia, 2012.
40. Abbas, M.; Majid, A.A.; Md. Ismail, A.I.; Rashid, A. Numerical method using cubic B-spline for a strongly coupled reaction-diffusion system. *PLoS ONE* **2014**, *9*, e83265. [CrossRef]
41. Rubin, S.G.; Graves, R.A.; Arjang, F. *A Cubic Spline Approximation for Problems in Fluid Mechanics*; NASA TR R-436; NASA: Washington, DC, USA, 1975.
42. Smith, G.D. *Numerical Solution of Partial Differential Equations: Finite Difference Methods*; Oxford University Press: Oxford, UK, 1985.
43. Akram, T.; Abbas, M.; Ali, A.; Iqbal, A.; Baleanu, D. A numerical approach of a time fractional reaction–diffusion model with a non-singular kernel. *Symmetry* **2020**, *12*, 1653. [CrossRef]
44. Akram, T.; Abbas, M.; Ali, A. A numerical study on time fractional fisher equation using an extended cubic b-spline approximation. *J. Math. Comput. Sci.* **2021**, *22*, 85–96. [CrossRef]
45. Iqbal, M.k.; Abbas, M.; Nazir, T.; Ali, N. Application of new quintic polynomial b-spline approximation for numerical investigation of kuramoto-sivashinsky equation. *Adv. Differ. Equ.* **2020**, *2020*, 558. [CrossRef]
46. Siddiqi, S.S.; Arshed, S. Quintic B-spline for the numerical solution of the good Boussinesq equation. *J. Egypt. Math. Soc.* **2014**, *22*, 209–213.

47. Poulin, J.R. *Calculating Infinite Series Using Parseval's Identity*; The University of Maine: Orono, ME, USA, 2020.
48. Taghipour, M.; Aminikhah, H. A B-Spline Quasi Interpolation Crank–Nicolson Scheme for Solving the Coupled Burgers Equations with the Caputo–Fabrizio Derivative. *Math. Probl. Eng.* **2021**, *2021*, 8837846. [CrossRef]