fractal and fractional

MDPI

Article

Quadrature Solution for Fractional Benjamin–Bona–Mahony–Burger Equations

Waleed Mohammed Abdelfattah [1], Ola Ragb [2], Mokhtar Mohamed [3,*], Mohamed Salah [2] and Abdelfattah Mustafa [4,5]

[1] College of Engineering, University of Business and Technology, Jeddah 23435, Saudi Arabia; w.abdelfattah@ubt.edu.sa
[2] Department of Engineering Mathematics and Physics, Faculty of Engineering, Zagazig University, Zagazig 44519, Egypt; ormohamed@eng.zu.edu.eg (O.R.); msalaheldin@zu.edu.eg (M.S.)
[3] Basic Science Department, Faculty of Engineering, Delta University for Science and Technology, Gamasa 11152, Egypt
[4] Department of Mathematics, Faculty of Science, Islamic University of Madinah, Madinah 42351, Saudi Arabia; amelsayed@mans.edu.eg
[5] Department of Mathematics, Faculty of Science, Mansoura University, Mansoura 35516, Egypt
* Correspondence: mokhtar.alsaidi@deltauniv.edu.eg

Abstract: In this work, we present various novelty methods by employing the fractional differential quadrature technique to solve the time and space fractional nonlinear Benjamin–Bona–Mahony equation and the Benjamin–Bona–Mahony–Burger equation. The novelty of these methods is based on the generalized Caputo sense, classical differential quadrature method, and discrete singular convolution methods based on two different kernels. Also, the solution strategy is to apply perturbation analysis or an iterative method to reduce the problem to a series of linear initial boundary value problems. Consequently, we apply these suggested techniques to reduce the nonlinear fractional PDEs into ordinary differential equations. Hence, to validate the suggested techniques, a solution to this problem was obtained by designing a MATLAB code for each method. Also, we compare this solution with the exact ones. Furthermore, more figures and tables have been investigated to illustrate the high accuracy and rapid convergence of these novel techniques. From the obtained solutions, it was found that the suggested techniques are easily applicable and effective, which can help in the study of the other higher-D nonlinear fractional PDEs emerging in mathematical physics.

Keywords: generalized Caputo; quadrature approach; discrete singular convolution; perturbation method; fractional nonlinear PDEs; Benjamin–Bona–Mahony–Burger equation

Citation: Abdelfattah, W.M.; Ragb, O.; Mohamed, M.; Salah, M.; Mustafa, A. Quadrature Solution for Fractional Benjamin–Bona–Mahony–Burger Equations. *Fractal Fract.* **2024**, *8*, 685. https://doi.org/10.3390/fractalfract8120685

Academic Editor: Wei-Shih Du

Received: 15 October 2024
Revised: 18 November 2024
Accepted: 21 November 2024
Published: 22 November 2024

1. Introduction

Numerous problems in the world can be determined via physical and mathematical models. It has been found that these models are basically connected to linear and non-linear PDEs, which can be utilized to represent numerous real-life phenomena, for instance, plasma physics, optical fiber, solid state physics, fluid mechanics, geochemistry, and chemical physics.

Nonlinear fractional partial differential equations have garnered significant attention due to their ability to model complex phenomena across various fields, including mathematical physics, plasma physics, population dynamics, electromagnetism, neutron point kinetics, acoustics, control and vibration, viscoelasticity, and fluid dynamics [1–12].

The Benjamin–Bona–Mahony–Burger (BBM-Burger) equation, a prominent model in ocean engineering, provides valuable insights into various wave phenomena. Its applications extend to acoustic waves in precious stones, hydromagnetic waves in plasma, thermodynamics, and acoustic-gravity waves in fluids. The BBM-Burger equation is particularly useful in the field of fluid dynamics, especially for modeling tsunami propagation from the ocean [1–5,13].

Benjamin et al. [13] presented Benjamin–Bona–Mahony equation (BBME) for the first time to model long waves of short amplitude in some nonlinear dispersive media. This equation can also express acoustic waves in inharmonic crystals, hydromagnetic waves in cold plasma, and acoustic–gravity waves in compressible fluids [13,14].

Nonlinear fractional PDEs do not have exact solutions in most problems. Thus, it is essential to advance efficient and accurate analytical and numerical techniques. Several techniques have been used to obtain analytical and numerical solutions for the fractional BBM equation, including the meshfree technique [15], Adomian decomposition technique [16], and finite element techniques in [17–20]. Ali Barati [21] analyzed a Sinc collocation technique for solving the time-fractional (BBME) equation. Yaro et al. [22] explained the solution of the space–time-fractional Zakhorov Kuznetsov BBME (ZKBBME) and the space–time-fractional symmetric regularized long wave (SRLWE) via the improved F-expansion technique. Kumar [23] used Lie symmetry and modified (G′/G)-expansion methods to examine the traveling wave solutions of a coupled BBM-KdV equation. Liu [24] demonstrated the approximate solution of the fractional nonlinear equations via a homotopy perturbation transformation technique (HPTM). Ray and Das [25] examined the BBM-Burger equation numerically and analytically using a reproducing kernel Hilbert space method. Dehghan et al. [26] explained the Legendre spectral element method (LSEM) for solving a nonlinear generalized (GBBMB) equation. Elmandouh and Fadhal [27] explored the effect of space-fractional and multiplicative noise on the analytic solutions of the space-fractional stochastic dispersive modified BBME. Javeed et al. [28] established a first integral technique to obtain analytic solutions for a space–time-fractional modified BBME and the coupled time-fractional Boussinesq–Burgers equation. Dehghan et al. [29] used forward-type finite difference and Kansa's method to obtain the solutions of the GBBM–Burger equation (GBBMBE). Oruç [30] solved the 1D and 2D versions of the GBBMBE via an algorithm depending on Lucas polynomials. Dehghan et al. [31] applied an element-free Galerkin approach to solve the 2D GBBMB equation. Hajiketabi et al. [32] concentrated on the high dimensional GBBMBE via a Lie-group technique depending on RBFs. Bayarassou [33] offered the 1D GBBMBE numerically through two high-order implicit difference approaches. Arora et al. [34] introduced a collocation technique depending on Hermit splines and a weighted finite difference scheme to solve the BBM-Burger equation. Islam et al. [35] examined the space–time-fractional modified BBME via the extended tanh technique, fractional generalized $(D_{\xi}{}^{\alpha}\, G'/G)$-expansion technique, and the Exp-function technique. Ege and Misirli [36] applied the modified Kudryashov technique to solve the space–time-fractional modified BBME and the space–time-fractional potential Kadomtsev–Petviashvili equation. Guner and Bekir [37] proposed an ansatz technique to compute the solutions of the space–time-fractional modified BBME, the time-fractional mKdVE, and the nonlinear fractional Zoomeron equation. Barati [21] analyzed a Sinc collocation technique to solve the time-fractional BBME, and the time variables were discretized via the finite forward difference procedure. Kapoor and Joshi [38] presented the numerical solution for the 1D nonlinear Burger's equation using a differential quadrature method based on a cubic uniform algebraic trigonometric tension B-spline. Also, Joshi et al. [39] proposed a novel numerical solution for 1D and 2D coupled nonlinear Schrödinger equations based on DQM. Castro López et al. [40] presented analytical solutions to a generalized Gross–Pitaevskii equation. Also, Gaussian solitary wave solutions were proposed for a nonlinear Schrödinger equation in [41–45].

In this paper, differential quadrature methods (DQM) based on polynomial (PDQM) and discrete singular convolution (DSCDQM) with Caputo and generalized Caputo types are used to construct a numerical solution of the space–time-fractional modified BBME and BBMBE.

The discrete sine collocation discrete differential quadrature method (DSCDQM) is a highly efficient numerical technique that offers several advantages over traditional methods [46–49]. By utilizing a minimal number of grid points and requiring less computational time, the DSCDQM can produce accurate and efficient solutions to complex problems.

Furthermore, its flexibility in choosing shape functions enhances its applicability to a wide range of equations.

To the best of the authors' knowledge, this research represents the first application of DSCDQM based on two specific shape functions, namely the regularized Shannon kernel (RSK) and the regularized Dirichlet kernel (RDK), to solve time-fractional modified Benjamin–Bona–Mahony (BBM) and Benjamin–Bona–Mahony–Burger (BBM-Burger) equations. Perturbation and iterative techniques were employed to linearize the nonlinear fractional PDEs. Subsequently, the discrete differential quadrature method (DDQM), DSCDQM-RSK, and DSCDQM-DK were applied using both Caputo and generalized Caputo fractional derivatives to transform the governing equations into linear algebraic equations.

A MATLAB code was developed to implement these methods and solve the equations for each approach, as illustrated in Figure 1. The accuracy and effectiveness of the proposed methods were validated by comparing their results with existing analytical and numerical solutions. Additionally, a parametric analysis was conducted to investigate the impact of the techniques on soliton solutions.

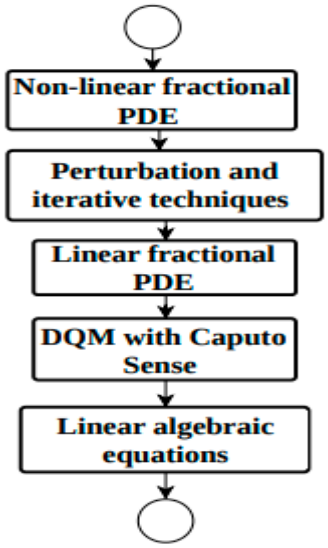


Figure 1. Numerical algorithm procedure solution.

2. Model Formulation of the Problem

In this investigation, we concentrate on the following two nonlinear equations, known as the space–time-fractional modified BBME and BBMBE, which are very important in the field of physics and natural sciences [35,36]:

2.1. Space–Time-Fractional Modified BBME [35,36]

$$c\frac{\partial^{\gamma,\delta}U(x,t)}{\partial t^{\gamma,\delta}} = -k\frac{\partial^{\gamma,\delta}U(x,t)}{\partial x^{\gamma,\delta}} + vkU^2(x,t)\frac{\partial^{\gamma,\delta}U(x,t)}{\partial x^{\gamma,\delta}} - k^3\frac{\partial^{\gamma,\delta}}{\partial x^{\gamma,\delta}}\left(\frac{\partial^{\gamma,\delta}}{\partial x^{\gamma,\delta}}\left(\frac{\partial^{\gamma,\delta}}{\partial x^{\gamma,\delta}}U(x,t)\right)\right) \quad in\ \Im \times (0,H], 0 < \gamma \leq 1 \tag{1}$$

where $\Im$ is the computational domain $\Im = [x_1, x_2] \subset \Re$, $(0, H]$ is the time interval. k, v and c are constants, where $v > 0$. U is function of x and t that describes acoustic gravitational waves in compressible fluids, hydromagnetic waves in cold plasma, and surface long waves in nonlinear dispersive media. γ and δ represent fractional order derivatives.

The boundary conditions can be shown for this problem as follows [35,36]:

$$P_1\,U + G_1\frac{\partial U}{\partial x} = Q_1(t), \qquad \text{at } (x_1, t) \tag{2}$$

$$P_2\, U + G_2 \frac{\partial U}{\partial x} = Q_2(t), \qquad \text{at } (x_2, t) \tag{3}$$

Furthermore, the initial condition of this problem can be demonstrated as follows [35,36]:

$$U(x, 0) = \Theta(x), \tag{4}$$

where P_i, G_i, Q_i, $(i = 1, 2)$, and $\Theta(x)$ are known functions.

2.2. Time-Fractional BBMBE [35,36]

$$\begin{aligned} \frac{\partial^{\gamma,\delta} U(x,t)}{\partial t^{\gamma,\delta}} = \frac{\partial^{\gamma,\delta} U_{xx}(x,t)}{\partial t^{\gamma,\delta}} - U_x(x,t) - U(x,t)U_x(x,t) \\ in\ \Im \times (0, H], 0 < \gamma \leq 1 \end{aligned} \tag{5}$$

The boundary and initial conditions subjected to this problem can be written as in the previous Equations (2)–(4).

3. Methods of Solution

This section describes a differential quadrature depending on three base functions (Lagrange, RSK, and RDK), as well as Caputo and generalized Caputo sense to solve the space–time-fractional modified BBME and BBMBE. Also, it gives the main steps of these methods. Furthermore, we discuss the perturbation and Iterative methods for treating the nonlinearity.

Now, we present a brief review of the DQM. The DQM involves estimating a derivative of a specific function $\Phi(x)$ via the linear summation of its components at various nodes of the problem domain [a, b]. This field can be simply divided into uniformly distributed finite nodes x_i $(i = 1 : n)$ with distance Δ, such that $a = x_1 < x_2 < \cdots < x_{n-1} < x_n = b$. The DQ discretization of the first and second derivatives at a node x_i is determined by the following Equation (6) [49]:

$$\left.\frac{\partial \Phi}{\partial x}\right|_{x=x_i} = \sum_{j=1}^{n} \psi_{ij}^{1}\, \Phi(x_j), \quad \left.\frac{\partial^2 \Phi}{\partial x^2}\right|_{x=x_i} = \sum_{j=1}^{n} \Psi_{ij}^{2}\, \Phi(x_j), \quad i, j = 1 : n \tag{6}$$

where Ψ_{ij}^{1} and Ψ_{ij}^{2} are the first and second weighting coefficients [49]. Since these weighting coefficients depend on the spatial grid spacing n and the choice of test functions, many researchers have used different test functions to create various kinds of DQMs [49–59].

3.1. Polynomial Differential Quadrature (PDQ) Technique

In this section, we introduce the polynomial differential quadrature (PDQ) technique. This technique is based on a Lagrange interpolation polynomial, which serves as a test function. The following reference provides further details on the PDQ technique [49].

$$\Phi(x_i) = \sum_{j=1}^{n} \frac{\prod_{k=1}^{n} (x_i - x_k)}{(x_i - x_j) \prod_{j=1, j\neq k}^{n} (x_j - x_k)} \Phi(x_j), \quad (i = 1 : n), \tag{7}$$

Hence, the weighting coefficients of the first derivative Ψ_{ij}^{1} can be given as follows [49]:

$$\Psi_{ij}^{1} = \begin{cases} \dfrac{1}{(x_i - x_j)} \prod\limits_{k=1, k\neq i,j}^{n} \dfrac{(x_i - x_k)}{(x_j - x_k)} & i \neq j \\ -\sum\limits_{j=1, j\neq i}^{n} \Psi_{ij}^{1} & i = j \end{cases} \tag{8}$$

Consequently, the weighting coefficients of the mth derivatives can be given by matrix multiplication, as follows [49]:

$$\left[\Phi_{ij}^{m}\right] = \left[\Phi_{ij}^{1}\right]\ \left[\Phi_{ij}^{m-1}\right]\ , (m = 2, 3, 4) \tag{9}$$

3.2. Description of Discrete Singular Convolution DQM (DSCDQM)

In the DSC method, a function $\Phi(x)$ and its mth-order derivative can be approximated via discretized singular kernels of the delta type [54–57], as follows:

$$\Phi^{m}(x_i) = \sum_{j=-M}^{M} \delta_{\alpha,\Delta}^{m}\,(x_i - x_j)\,\Phi(x_j) \;=\; \sum_{j=-M}^{M} \Psi_{j}^{m}\,\Phi(x_j),\ (i = -n,\, n),\ n > M \tag{10}$$

where $\delta_{\alpha,\Delta}^{m}\,(x_i - x_j)$ is a DSC kernel and M is the bandwidths for estimating the function. Δ is the spacing between two adjacent points, while α is a parameter used in regularization.

Although many kernels may be used in the DSC method [56–59], the DL, RS, and RD kernels are used in this work [54–57].

1. Regularized Shannon kernel (RSK)

Here, we use regularized Shannon as the test function. Therefore, depending on the DQM (Equation (11)), the weighting coefficients of the mth-order derivatives at point xi are determined as follows [54–57]:

$$\Phi(x_i) = \sum_{j=-M}^{M} \left\langle \frac{\sin\left[\dfrac{\pi(x_i - x_j)}{\Delta}\right]}{\dfrac{\pi(x_i - x_j)}{\Delta}}\ \exp\left(\frac{-(x_i - x_j)^2}{2\alpha^2}\right) \right\rangle \Phi(x_j),\ (i = -n : n),\ \alpha = (f \times \Delta\,) > 0 \tag{11}$$

The weighting coefficients Ψ_{ij}^{1} and Ψ_{ij}^{2} based on the differentiation Equation (11) are explained using DSCDQM-RSK as follows [54–57]:

$$\Psi_{ij}^{1} = \begin{cases} \dfrac{(-1)^{i-j}}{\Delta(i-j)}\exp\left(-\Delta^2\left(\dfrac{(i-j)^2}{2\alpha^2}\right)\right), & i \neq j \\ 0 & i = j \end{cases},\quad \Psi_{ij}^{2} = \begin{cases} \left(\dfrac{2(-1)^{i-j+1}}{\Delta^2(i-j)^2} + \dfrac{1}{\alpha^2}\right)\exp\left(-\Delta^2\left(\dfrac{(i-j)^2}{2\alpha^2}\right)\right), & i \neq j \\ -\dfrac{1}{\alpha^2} - \dfrac{\pi^2}{3\Delta^2} & i = j \end{cases} \tag{12}$$

2. Regularized Dirichlet kernel (RDK)

Here, we use regularized Dirichlet as the test function. Therefore, depending on the DQM (Equation (13)), the weighting coefficients of the mth-order derivatives at point x_i are defined as follows [54–57]:

$$\Phi(x_i) = \sum_{j=-M}^{M} \left\langle \frac{\sin\left[\dfrac{\pi(x_i - x_j)}{\Delta}\right]}{(2T+1)\tan\left[\dfrac{\pi(x_i - x_j)}{\Delta(2T+1)}\right]}\ \exp\left(\frac{-(x_i - x_j)^2}{2\alpha^2}\right) \right\rangle \Phi(x_j),\ (i = -n : n),\ \alpha = (f \times \Delta\,) > 0 \tag{13}$$

The weighting coefficients Ψ_{ij}^{1} and Ψ_{ij}^{2} based on the differentiation Equation (13) are explained using DSCDQM-RDK as follows [54–57]:

$$\Psi_{ij}^{1} = \begin{cases} \dfrac{\pi(-1)^{i-j}}{\Delta(2T+1)\tan\left[\frac{\pi(i-j)}{\Delta(2T+1)}\right]}\exp\left(-\Delta^2\left(\dfrac{(i-j)^2}{2\alpha^2}\right)\right), & i \neq j \\ 0 & i = j \end{cases}, \tag{14}$$

$$\Psi_{ij}^{2} = \begin{cases} \left(\frac{2\pi^2(-1)^{i-j+1}}{\Delta^2(2T+1)^2 \sin^2\left[\frac{\pi(i-j)}{(2T+1)}\right]} + \frac{2\pi(i-j)(-1)^{i-j+1}}{\alpha^2(2T+1)\tan\left[\frac{\pi(i-j)}{(2T+1)}\right]} \right) \exp(-\Delta^2(\frac{(i-j)^2}{2\alpha^2})), & i \neq j \\ -\frac{1}{\alpha^2} - \frac{\pi^2}{3\Delta^2} & i = j \end{cases} \tag{15}$$

where T is a parameter, if $T \to \infty$ RDK was converted to RSK.

Next, we employed the numerical techniques PDQM, DSCDQM-RSK, and DSCDQM-RDK with a fractional derivative; we discuss Caputo and generalized Caputo definitions, which are the most novel definitions.

3.3. Generalized Caputo-Kind Fractional Derivative

Fractional-order DEs extend more accurate models of real-world problems and are suitable for describing many events in porous media or on unsmooth boundaries. Therefore, it is found that fractional calculus has developed greatly in the literature over the past few decades due to its wide use in different disciplines of science and engineering. Several fractional derivative (FD) types exist, including Riemann–Liouville, Caputo, Hadamard, Weyl, Grunwald–Letnikov, and Marchaud. Now, we provide some definitions of the Caputo FD that will be used in the sequel to the work, as follows [57–66]:

- Caputo's Fractional Derivative

This section briefly summarizes Caputo's FD, which is based on the Riemann–Liouville FD [51–55].

Assume that $\gamma \in R^+$. If m is a positive integer, then $m-1 < \gamma \leq m$. The Riemann–Liouville FD of a function $\Phi(x)$ of order γ is written as follows:

$$D_a^{\gamma}\Phi(x) = \frac{1}{\Gamma_{(m-\gamma)}} \frac{d^m}{dx^m} \int_a^x (x-t)^{m-\gamma-1}\, \Theta^m(t)\, dt, \tag{16}$$

where $D_a^{\gamma}\Phi(x)$ is the fractional derivative of $\Phi(x)$ and a is the integration lower limit.

Then Caputo's FD of order γ is explained as follows:

$$D_a^{\gamma}\Phi(x) = \begin{cases} \frac{1}{\Gamma_{(m-\gamma)}} \int_a^x (x-t)^{m-\gamma-1}\, \Phi^m(t)\, dt, & m-1 < \gamma < m \\ \frac{d^m\Phi}{dx^m} & m = \gamma \end{cases} \tag{17}$$

We define some properties of the Caputo derivative as follows:

$$D_a^{\gamma}(c\, f(x) + dg(x)) = c\, D_a^{\gamma} f(x) + d\, D_a^{\gamma} g(x) \tag{18}$$

$$D_a^{\gamma} c = 0, \tag{19}$$

$$D_a^{\gamma} x^p = \begin{cases} 0 & p < \gamma \\ \frac{\Gamma(p+1)}{\Gamma(p-\gamma+1)}\, x^{p-\gamma} & otherwise \end{cases}, \quad p = (0, 1, 2 \ldots) \tag{20}$$

where $f(x)\ and\ g(x) : \mathbb{R}+ \to \mathbb{R}$ represent arbitrary functions, such as transcendental and polynomials functions, while c and d are constant.

Also, the generalized Caputo's fractional derivative is written as follows:

Fractional differential operators, due to their non-local nature, are particularly well-suited for modeling systems with memory effects [62–65]. Various forms of non-locality exist, prompting researchers to explore fractional operators as a means to capture the hidden complexities of real-world non-local phenomena.

We provide the following definition of the generalized Caputo fractional derivative of any order, as presented in the referenced literature [62–65]:

$$D_{a_+}^{\gamma,\delta}\Phi(x) = \frac{\delta^{\gamma-m+1}}{\Gamma(m-\gamma)}\int_a^x t^{\delta-1}(x^\delta - t^\delta)^{m-\gamma-1}\left(t^{1-\delta}\frac{d}{dt}\right)^m \Phi(t)\,dt, \quad m-1<\gamma<m,\ \delta>0,\ a\geq 0 \tag{21}$$

Consequently, the solution of Equation (21) can be taken as follows [56–60]:

$$D_{a_+}^{\gamma,\delta}(x^\delta - t^\delta)^m = \delta^\gamma \frac{\Gamma(m+1)}{\Gamma(m-\gamma+1)}\left(x^\delta - a^\delta\right)^{m-\gamma} \tag{22}$$

Now, we demonstrate the novel numerical technique by combining the Caputo's FD and generalized Caputo's FD in Equation (22) with Equation (6) of PDQM, DSCDQM-RSK, and DSCDQM-RDK to compute the weighting coefficients Ψ_{ij}^{γ}, $\psi_{ij}^{\gamma,\delta}$ for $\gamma \in (0,1]$ and $\delta > 0$, as follows [49,52]:

$$D^\gamma\Phi(x) = \begin{cases} \frac{1}{\Gamma(1-\gamma)}\int_a^x (x-t)^{-\gamma}\Phi'(t)\,dt = \sum_{j=1}^{n}\Psi_{ij}^{\gamma}\,\Phi(x_j,t), & 0<\gamma<1 \\ & i=1:n \\ \sum_{j=1}^{n}\Psi_{ij}^{1}\,\Phi(x_j,t) & \gamma=1 \end{cases} \tag{23}$$

$$D_a^{\gamma,\delta}\Phi_{(x)} = \begin{cases} \frac{\delta^\gamma}{\Gamma(1-\gamma)}\int_a^x t^{\delta-1}[x^\delta - t^\delta]^{-\gamma}\left(t^{1-\delta}\frac{d}{dt}\right)'\Phi(t)\,dt = \sum_{j=1}^{n}\Psi_{ij}^{\gamma,\delta}\,\Phi(x_j,t), & 0<\gamma\leq 1\ ,\ \delta>0 \\ & i=1:n \\ \sum_{j=1}^{n}\Psi_{ij}^{1}\,\Phi(x_j,t) & \gamma=\delta=1 \end{cases} \tag{24}$$

Then, the weighting coefficients Ψ_{ij}^{γ}, $\psi_{ij}^{\gamma,\delta}$ are computed as follows [49,52]:

$$\Psi_{\mathrm{ij}}^{\gamma} = \mathrm{A}^{1-\gamma}\,\Psi_{\mathrm{ij}}^{1} - \frac{\Psi_{1,\mathrm{j}}^{1}}{\Gamma(2-\gamma)}\,(\mathrm{x}-\mathrm{a})^{1-\gamma},\quad \mathrm{A_{ij}} = \Psi_{\mathrm{ij}}^{1} - \Psi_{1\mathrm{j}}^{1} \tag{25}$$

$$\Psi_{\mathrm{ij}}^{\gamma,\delta} = \mathrm{A}^{1-\gamma}\delta^\gamma\,\Psi_{\mathrm{ij}}^{1} + \frac{\delta^\gamma\Psi_{1,\mathrm{j}}^{1}}{\Gamma(2-\gamma)}\,(\mathrm{x}^\delta-\mathrm{a}^\delta)^{1-\gamma},\quad \mathrm{A_{ij}} = \Psi_{\mathrm{ij}}^{1} - \Psi_{1\mathrm{j}}^{1} \tag{26}$$

Equations (25) and (26) can be explained as follows [57,60]:

$$D_x^\gamma\Phi(x) = J_a^{1-\gamma}\left(\frac{\partial\Phi(x)}{\partial x}\right) = J_a^{1-\gamma}(\Phi(x)) = J_a^{1-\gamma}(\Phi'(x)-\Phi'(a)), \tag{27}$$

$$D_x^\gamma\Phi(x) = J_a^{1-\gamma}\left(\frac{\partial\Phi(x)}{\partial x}\right) = \sum_{j=1}^{n}\Psi_{ij}^{\gamma}\Phi(x_j,t) = \left(A^{1-\gamma}\Psi_{ij}^{1} - \frac{d}{\Gamma(2-\gamma)}\,(x-a)^{1-\gamma}\right)\Phi \tag{28}$$

$$\text{Then } \Psi_{ij}^{\gamma} = A^{1-\gamma}\Psi_{ij}^{1} - \frac{d}{\Gamma(2-\gamma)}\,(x-a)^{1-\gamma} \tag{29}$$

Equations (27) and (28) can be proved as follows:

$$\Phi'(a) = d\,\Phi(a),\ \ d = \Psi^1_{1j},\ J^{\gamma}\Phi'(a) = d\,J^{\gamma}\Phi(a) = d\,\frac{\Phi(a)}{\Gamma(\gamma)}\int_a^x (x-t)^{\gamma-1}\,dt = \frac{\Phi(a)}{\Gamma(\gamma+1)}d\,(x-a)^{\gamma}, \tag{30}$$

Consequently, Equation (31) is as follows:

$$J_a^{1-\gamma}\Phi'(a) = \frac{\Phi(a)}{\Gamma(2-\gamma)}d\,(x-a)^{1-\gamma}, \tag{31}$$

Also, Equation (32) is as follows:

$$\int_a^x \Phi(t)\,dt = \sum_{j=1}^{n}(\Psi^1_{ij} - \Psi^1_{1j})\Phi(x_j,t), \quad \mathrm{A}_{ij} = \Psi^1_{ij} - \Psi^1_{1j},\ \ i = 1:n \tag{32}$$

Then, Equation (33) is as follows:

$$J^1\Phi(x) = \int_a^x \Phi(t)\,dt = \mathrm{A}\,\Phi(x)\ \ \Rightarrow\ J^2\Phi(x) = \int_a^x\int_a^x \Phi(t)\,dt = \int_a^x (x-t)\Phi(t)\,dt = \mathrm{A}^2\Phi(x) \tag{33}$$

Furthermore, Equation (34) is as follows:

$$J^{\gamma}\Phi(x) = \mathrm{A}^{\gamma}\Phi(x)\ \ \Rightarrow\ J^{1-\gamma}\Phi'(x) = \mathrm{A}^{1-\gamma}\Psi^1_{ij}\Phi(x) \tag{34}$$

Due to the nonlinearity of these problems, we can apply perturbation and iterative techniques as follows.

3.4. Perturbation Technique [65]

We can solve the modified BBM Equation (1) via the perturbation technique by assuming that Equation (35) is as follows:

$$\mathrm{U} = \mathrm{U}_o + \eta\mathrm{U}_1 + \eta^2\mathrm{U}_2 + \cdots \eta^{\mathrm{n}}\mathrm{U}_{\mathrm{n}} \tag{35}$$

where $\mathrm{U}_o, \mathrm{U}_1$ and U_2 are unknown functions. η is a perturbation parameter.

The MBBM Equation (1) can be transformed into the following series of linear problems after substituting from Equation (35), as follows:

$$\begin{aligned} c\frac{\partial^{\gamma,\delta}(\mathrm{U}_o+\eta\mathrm{U}_1+\eta^2\mathrm{U}_2+\cdots\eta^{\mathrm{n}}\mathrm{U}_{\mathrm{n}})}{\partial \mathrm{t}^{\gamma,\delta}} = \ & -k\frac{\partial^{\gamma,\delta}(\mathrm{U}_o+\eta\mathrm{U}_1+\eta^2\mathrm{U}_2+\cdots\eta^{\mathrm{n}}\mathrm{U}_{\mathrm{n}})}{\partial \mathrm{x}^{\gamma,\delta}} + vk\eta\,(\mathrm{U}_o+\eta\mathrm{U}_1+\eta^2\mathrm{U}_2+\cdots\eta^{\mathrm{n}}\mathrm{U}_{\mathrm{n}})^2 \\ & \frac{\partial^{\gamma,\delta}(\mathrm{U}_o+\eta\mathrm{U}_1+\eta^2\mathrm{U}_2+\cdots\eta^{\mathrm{n}}\mathrm{U}_{\mathrm{n}})}{\partial \mathrm{x}^{\gamma,\delta}} - k^3\,\frac{\partial^{\gamma,\delta}}{\partial \mathrm{x}^{\gamma,\delta}}\,(\frac{\partial^{\gamma,\delta}}{\partial \mathrm{x}^{\gamma,\delta\delta}}\,(\frac{\partial^{\gamma,\delta}}{\partial \mathrm{x}^{\gamma,\delta}}\,(\mathrm{U}_o+\eta\mathrm{U}_1+\eta^2\mathrm{U}_2+\cdots\eta^{\mathrm{n}}\mathrm{U}_{\mathrm{n}}))) \end{aligned} \tag{36}$$

Also, the boundary and initial conditions can be exhibited after the substituting from Equations (2)–(4), as follows:

$$\begin{aligned} & \Big[\mathrm{P}_1\mathrm{U}_0(\mathrm{x}_1,\mathrm{t}) + \mathrm{G}_1\tfrac{\partial \mathrm{U}_0}{\partial \mathrm{x}}\Big|\ (\mathrm{x}_1,\mathrm{t})\ \Big] + \eta\Big[\mathrm{P}_1\mathrm{U}_1(\mathrm{x}_1,\mathrm{t}) + \mathrm{G}_1\tfrac{\partial \mathrm{U}_1}{\partial \mathrm{x}}\Big|\ (\mathrm{x}_1,\mathrm{t})\ \Big] \\ & +\eta^2\Big[\mathrm{P}_1\mathrm{U}_2(\mathrm{x}_1,\mathrm{t}) + \mathrm{G}_1\tfrac{\partial \mathrm{U}_2}{\partial \mathrm{x}}\Big|\ (\mathrm{x}_1,\mathrm{t})\ \Big] + \ldots\eta^{\mathrm{n}}\Big[\mathrm{P}_1\mathrm{U}_{\mathrm{n}}(\mathrm{x}_1,\mathrm{t}) + \mathrm{G}_1\tfrac{\partial \mathrm{U}_{\mathrm{n}}}{\partial \mathrm{x}}\Big|\ (\mathrm{x}_1,\mathrm{t})\ \Big] = Q_1(\mathrm{t}) \end{aligned} \tag{37}$$

$$\begin{aligned} & \Big[\mathrm{P}_1\mathrm{U}_0(\mathrm{x}_1,\mathrm{t}) + \mathrm{G}_1\tfrac{\partial \mathrm{U}_0}{\partial \mathrm{x}}\Big|\ (\mathrm{x}_2,\mathrm{t})\ \Big] + \eta\Big[\mathrm{P}_1\mathrm{U}_1(\mathrm{x}_2,\mathrm{t}) + \mathrm{G}_1\tfrac{\partial \mathrm{U}_1}{\partial \mathrm{x}}\Big|\ (\mathrm{x}_2,\mathrm{t})\ \Big] \\ & +\eta^2\Big[\mathrm{P}_1\mathrm{U}_2(\mathrm{x}_2,\mathrm{t}) + \mathrm{G}_1\tfrac{\partial \mathrm{U}_2}{\partial \mathrm{x}}\Big|\ (\mathrm{x}_2,\mathrm{t})\ \Big] + \ldots\eta^{\mathrm{n}}\Big[\mathrm{P}_1\mathrm{U}_{\mathrm{n}}(\mathrm{x}_2,\mathrm{t}) + \mathrm{G}_1\tfrac{\partial \mathrm{U}_{\mathrm{n}}}{\partial \mathrm{x}}\Big|\ (\mathrm{x}_2,\mathrm{t})\ \Big] = Q_2(\mathrm{t}) \end{aligned} \tag{38}$$

$$U_0(x,0) + \eta U_1(x,0) + \eta^2 U_2(x,0) + \ldots \eta^n U_n(x,0) = \Theta(x) \tag{39}$$

Then, by equating the coefficient of $\eta^0, \eta^1, \eta^2, \ldots \eta^n$ to obtain the final value of U, Equation (40) can be written as follows:

$$U_{numerical} = \lim_{\eta \to 1} [U_o + \eta U_1 + \eta^2 U_2 + \cdots \eta^n U_n] \tag{40}$$

Hence, to ensure the convergence of the results [67], we applied the condition test to the previous series in Equation (36) as follows to produce Equation (41):

$$\left| \frac{U_{i+1}}{U_i} \right| < 1 \quad \text{where} \, i = 0, 1, \ldots n-1 \tag{41}$$

3.5. Iterative Quadrature [68]

We solved the following iterative system as follows:

$$\begin{aligned} c \frac{\partial^{\gamma,\delta} U_{i+1}(x,t)}{\partial t^{\gamma,\delta}} = \quad & -k \frac{\partial^{\gamma,\delta} U_{i+1}(x,t)}{\partial x^{\gamma,\delta}} + vk \, {U_i}^2{}_{(x,t)} \frac{\partial^{\gamma,\delta} U_{i+1}(x,t)}{\partial x^{\gamma,\delta}} - \\ & k^3 \frac{\partial^{\gamma,\delta}}{\partial x^{\gamma,\delta}} \left(\frac{\partial^{\gamma,\delta}}{\partial x^{\gamma,\delta}} \left(\frac{\partial^{\gamma,\delta}}{\partial x^{\gamma,\delta}} U_{i+1}(x,t) \right) \right) \, , i = 0, 1, 2, \ldots \end{aligned} \tag{42}$$

Equation (42) was subject to previous boundary and initial conditions.

Consequently, BBMB equations can be solved via perturbation and iterative techniques, as with the MBBM Equation (5).

4. Numerical Results

In this section, we show the novelty of the suggested techniques by solving two nonlinear fractional problems. The suggested techniques are polynomial (PDQM) [51], DSCDQM—RSK, and DSCDQM—RDK [54–57] with the generalized Caputo's and Caputo type. Also, perturbation and iterative quadrature strategies are used to overcome the nonlinearity problems.

To validate the obtained results, first we compare these results with the exact results [35,36] to ensure the efficiency and accuracy of the suggested techniques. Second, we evaluate the stability and convergence of the obtained solutions by calculating the L_∞, root mean square (RMS), and L_2 errors as follows [57]:

$$L_\infty \text{ Error} = \max_{1 \le i \le N_x} \left| U_{numerical}(x_i, t_j) - U_{exact}(x_i, t_j) \right| \tag{43}$$

$$\text{RMS Error} = \sqrt{\left[\sum_{i,j=1}^{n_x, n_t} \left(U_{numerical}(x_i, t_j) - U_{exact}(x_i, t_j) \right)^2 \right] \Big/ (n_x \, n_t)} \tag{44}$$

$$L_2 \text{ Error} = \sqrt{\Delta x \, \Delta t \sum_{i,j=1}^{n_x, n_t} \left(U_{numerical}(x_i, t_j) - U_{exact}(x_i, t_j) \right)^2} \tag{45}$$

4.1. Consider a 1D Problem of MBBME Along the x-Direction as Follows

We applied the novel numerical technique on a fractional modified BBME by substituting the Equation (21) into Equation (1). However, the modified BBM equations are converted into nonlinear algebraic equations as follows:

$$\begin{aligned} c \sum_{j=1}^{n_t} \Psi_{tij}^{\gamma,\delta} U(x, t_j) = \quad & -k \sum_{i=1}^{n_x} \Psi_{xij}^{\gamma,\delta} U(x_i, t) + vk U^2{}_{(x_i,t_j)} \sum_{i=1}^{n_x} \Psi_{xij}^{\gamma,\delta} U(x_i, t) - \\ & k^3 \sum_{i=1}^{n_x} \Psi_{x_{ij}}^{\gamma,\delta} \left(\sum_{i=1}^{n_x} \Psi_{x_{ij}}^{\gamma,\delta} \left(\sum_{i=1}^{n_x} \Psi_{x_{ij}}^{\gamma,\delta} U_{(x_i,t)} \right) \right), \quad i = (1, n_x) \, , \; j = (1, n_t) \end{aligned} \tag{46}$$

Then, by subsituting Equation (21) into Equations (2)–(4), the boundary conditions and initial condition can be illustrated as follows:

$$P_1\,U\,(x_1,t)+G_1\sum_{i=1}^{n_x}\Psi^1_{1j}U\,(x_1,t)=Q_1(t) \tag{47}$$

$$P_2\,U(x_2,t)+G_2\sum_{i=1}^{n_x}\Psi^1_{2j}U\,(x_2,t)=Q_2(t)\,, \tag{48}$$

$$U(x_i,0)=\Theta(x_i), \tag{49}$$

where the values of $G_1 = G_2 = 0$, $P_1 = P_2 = 1$ and $\delta = 1$. Equations (50)–(52) are as follows:

$$Q_1(t)=\sqrt{\frac{3}{2kv}}k^{3/2}\tanh[\frac{kx_1{}^{\gamma}}{2\Gamma(1+\gamma)}+\frac{ct^{\gamma}}{2\Gamma(1+\gamma)}+\frac{\varepsilon}{2}], \tag{50}$$

$$Q_2(t)=\sqrt{\frac{3}{2kv}}k^{3/2}\tanh[\frac{kx_2{}^{\gamma}}{2\Gamma(1+\gamma)}+\frac{ct^{\gamma}}{2\Gamma(1+\gamma)}+\frac{\varepsilon}{2}],\quad 0\le t\le H \tag{51}$$

$$\Theta(x)=\sqrt{\frac{3}{2kv}}k^{3/2}\tanh[\frac{kx_1{}^{\gamma}}{2\Gamma(1+\gamma)}+\frac{\varepsilon}{2}],\quad x_1\le x\le x_2,\quad c=0.5k(-2+k^2) \tag{52}$$

In addition, the exact solution for the modified BBME can be given as follows [35]:

$$U_{exact}(x,t)=\sqrt{\frac{3}{2kv}}k^{3/2}\tanh[\frac{kx^{\gamma}}{2\Gamma(1+\gamma)}+\frac{ct^{\gamma}}{2\Gamma(1+\gamma)}+\frac{\varepsilon}{2}],\quad t>0,\quad x_1\le x\le x_2 \tag{53}$$

Table 1 displays the accuracy of PDQM at various values of γ and different grid sizes from (4 × 10) to (25 × 10) compared with exact values using perturbation and iterative quadrature techniques. PDQM with iterative quadrature is more accurate than the perturbation method at grid size (10 × 10). Iterative quadrature is more efficient than the perturbation method in terms of the calculating CPU time, which is equal 0.096258 s.

Table 1. Numerical solutions using PDQM via two strategies with various grid sizes of $n_x \times 6$ and different values of γ at $t = 0.2$, $x = 0.333$, $\delta = 1$.

n_x	PDQM + Perturbation			PDQM + Iterative		
	γ=1	γ=0.97	γ=0.95	γ=1	γ=0.97	γ=0.95
4	0.33410	0.33480	0.33525	0.32034	0.33251	0.33283
7	0.33452	0.33502	0.33582	0.33124	0.33502	0.33612
10	0.33541	0.33571	0.33601	0.33254	0.33752	0.33798
13	0.33554	0.33612	0.33675	0.33401	0.33791	0.33851
16	0.33587	0.33695	0.33725	0.33491	0.33810	0.33881
19	0.33588	0.33702	0.33790	0.33551	0.33817	0.33925
22	0.33601	0.33751	0.33851	0.33607	0.33831	0.33934
25	0.33608	0.33801	0.33908	0.33615	0.33897	0.33955
Exact [35]	0.33610	0.33803	0.33936	0.33610	0.33803	0.33936
CPU time	0.183610 s at 10 × 10			0.096258 s at 10 × 10		

Table 2 presents the influence of different parameters and grid sizes from size (9 × 6) to size (11 × 6) on the obtained results by DSCDQ-RSK and RDK methods combined with perturbation and iterative quadrature schemes. Thus, it is found that the obtained results from the DSCDQ-RSK method closely match the exact solutions at the parameters $M = 2$, $\alpha = 1 \times \Delta$ at $n_x = 10$, $n_t = 6$. The RDK results are also in good agreement with the exact

solutions and have high accuracy solutions at $M = 1$, $\alpha = 1 \times \Delta$ at $n_x = 10$, $n_t = 6$ and the parameter $T = 10$. When calculating the CPU time for RSK and RDK to prove the efficiency of these methods, it is remarkable the computation time for RSK is moderately less than that of RDK.

Table 2. Numerical solutions using the DSCDQM-RSK and RDK methods with various grid sizes, bandwidths $(2M + 1)$, and regularization parameter α at $\gamma = 1,\ \delta = 1,\ t = 1,\ v = k = \varepsilon = 1,\ x = 0.44$.

Grid Size	Bandwidth	DSC-RSK + Perturbation				DSC-RDK + Iterative			
		α				α			
	$2M+1$	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ
9	3	-	-	-	-	0.26846	0.26871	0.26877	0.26985
	5	0.26808	0.26811	0.26821	0.26821	0.26846	0.26871	0.26877	0.26985
	7	0.26808	0.26811	0.26821	0.26821	0.26846	0.26871	0.26877	0.26985
	9	0.26808	0.26811	0.26821	0.26821	0.26846	0.26871	0.26877	0.26985
	11	0.26808	0.26811	0.26821	0.26821	0.26846	0.26871	0.26877	0.26985
	15	0.26808	0.26811	0.26821	0.26821	0.26846	0.26871	0.26877	0.26985
	19	0.26808	0.26811	0.26821	0.26821	0.26846	0.26871	0.26877	0.26985
10	3	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	5	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	7	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	9	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	11	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	15	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	19	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
11	3	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	5	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	7	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	9	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	11	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	15	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
	19	0.26709	0.26813	0.26829	0.26849	0.26814	0.26819	0.26822	0.26838
CPU Time		0.095582 s. at $n_x = 10$, $M = 2$				0.096531 s. at $n_x = 10$, $M = 1$			
Exact [35]		0.26834							

Table 3 demonstrates the effect of T on the accuracy of the results for the DSC-RDK method. It is noted that the obtained results are characterized by rapid convergence and accurate results when the values parameter $T \geq 10$.

Table 3. Numerical solutions by the DSC-RDK method with various parameters (T) at $M = 1$, $\alpha = (1 \times \Delta)$ and $\gamma = 1,\ \delta = 1,\ t = 1,\ v = k = \varepsilon = 1,\ x = 0.44$.

Grid Points	T						
n_x	1	5	10	15	20	30	40
10	0.26820	0.26821	0.26823	0.26823	0.26823	0.26823	0.26823
11	0.26820	0.26821	0.26823	0.26823	0.26823	0.26823	0.26823
12	0.26820	0.26821	0.26823	0.26823	0.26823	0.26823	0.26823

Table 3. *Cont.*

Grid Points	T						
n_x	**1**	**5**	**10**	**15**	**20**	**30**	**40**
13	0.26820	0.26821	0.26823	0.26823	0.26823	0.26823	0.26823
14	0.26820	0.26821	0.26823	0.26823	0.26823	0.26823	0.26823
15	0.26820	0.26821	0.26823	0.26823	0.26823	0.26823	0.26823
	Exact [35]				0.26834		
	CPU Time				0.072366 s at $n_x = 10$		

Table 4 measures L_∞ error norms for three kernels, namely PDQM, DSC-RSK, and DSC-RDK, at different times and different values of γ to determine the reliability and stability of these methods. The RDK has $L_\infty \leq 2.99 \times 10^{-4}$ at grid size 10×10 and time = 10 s. RSK achieves accurate results with $L_\infty \leq 3.62 \times 10^{-4}$ at size (10×10) and time = 10 s. Also, PDQM achieves accurate results with $L_\infty \leq 7.8 \times 10^{-3}$ at size (10×10) and time = 10 s.

Table 4. L∞ error norms in [0, 1] at different times when $n_x \times n_t = 10 \times 6$ and $\delta = 1$,$\alpha = (1 \times \Delta)$.

Time	γ=1			γ=0.97		
	PDQM	**DSC-RSK** $M = 2$	**DSC-RDK** $M = 1$	**PDQM**	**DSC-RSK** $M = 2$	**DSC-RDK** $M = 1$
0.3	5.1778×10^{-4}	3.5078×10^{-4}	2.9011×10^{-4}	5.7120×10^{-4}	3.6210×10^{-4}	2.9801×10^{-4}
0.5	8.0243×10^{-4}	3.5241×10^{-4}	2.9021×10^{-4}	9.1042×10^{-4}	3.6201×10^{-4}	2.9811×10^{-4}
1	1.5487×10^{-3}	3.5277×10^{-4}	2.9021×10^{-4}	2.4017×10^{-3}	3.6251×10^{-4}	2.9724×10^{-4}
2	3.5518×10^{-3}	3.5188×10^{-4}	2.9005×10^{-4}	3.9118×10^{-3}	3.6280×10^{-4}	2.9880×10^{-4}
5	5.5001×10^{-3}	3.5001×10^{-4}	2.9011×10^{-4}	5.5289×10^{-3}	3.6291×10^{-4}	2.9818×10^{-4}
7	6.1901×10^{-3}	3.5002×10^{-4}	2.9003×10^{-4}	7.0991×10^{-3}	3.6292×10^{-4}	2.9921×10^{-4}
10	7.15410×10^{-3}	3.5012×10^{-4}	2.9015×10^{-4}	7.80662×10^{-3}	3.6294×10^{-4}	2.9912×10^{-4}

Table 5 shows the effect of generalized Caputo fractional derivatives γ, δ with the suggested techniques for the results obtained at $\delta = 1, 0.9$, $\gamma = 0.9, 0.8, 0.7$, $0 \leq t \leq 1$ and $x = 0.55$. The very good agreement among the obtained solutions with the exact ones [35] verifies the capability of these techniques to deal with the modified BBM problem. Also, the numerical solution U increases with increasing time and decreases with γ.

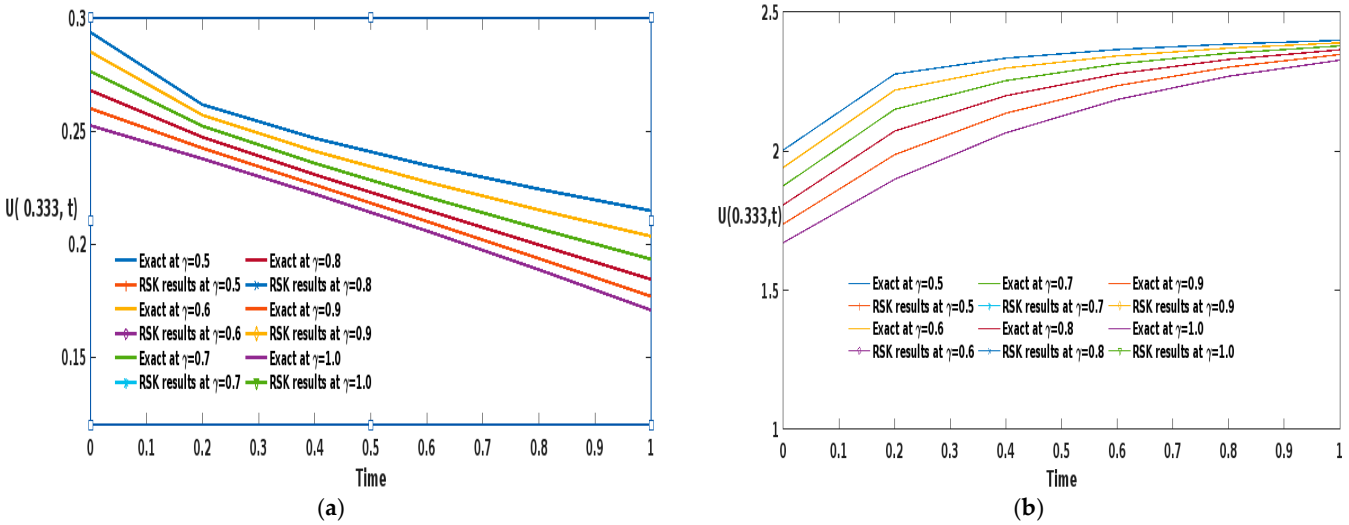
Table 5. Numerical solution (U) in [0, 1] at different times and derivative orders γ, δ when $n_x \times n_t = 10 \times 6$ and $x = 0.5556$, $M = 2$, $\alpha = (1 \times \Delta)$.

δ	γ	**Method**	**Time = 0.2**	**Time = 0.4**	**Time = 0.6**	**Time = 0.8**	**Time = 1**
1	0.9	PDQM	0.389721295	0.3680463518	0.3378722832	0.3346095	0.3056576
		RSK	0.3876536321	0.3640845129	0.34574756900	0.326278013	0.3056760
		RDK	0.3870390159	0.3619246792	0.34522989232	0.326552839	0.3056886
		Exact [35]	0.3872035353	0.36711214584	0.34683211164	0.326103456	0.3048345
	0.8	PDQM	0.3910653065	0.3753397944	0.35308995183	0.335620599	0.3180235
		RSK	0.3901965632	0.37030983171	0.35153027266	0.330488302	0.3138299
		RDK	0.3909431115	0.37069452570	0.35823577077	0.336317366	0.3199452
		Exact [35]	0.3930743748	0.37260909741	0.35297863310	0.333601339	0.3142535

Table 5. *Cont.*

δ	γ	Method	Time = 0.2	Time = 0.4	Time = 0.6	Time = 0.8	Time = 1
1	0.7	PDQM	0.3906530165	0.37012073090	0.37684786559	0.352215195	0.3128846
		RSK	0.3986530659	0.37702073090	0.35269978635	0.341211190	0.3286423
		RDK	0.3986530659	0.37702073090	0.35699786359	0.341211190	0.3286423
		Exact [35]	0.3981059865	0.37769699000	0.35912585548	0.341445249	0.3242833
0.9	0.9	PDQM	0.3663380120	0.34596421740	0.3176178900	0.314533214	0.2873180
		RSK	0.3643942101	0.34220197239	0.32501543003	0.301806701	0.2587335
		RDK	0.3638174100	0.34021507209	0.3221424516	0.302578696	0.2847347
	0.8	PDQM	0.3675134601	0.35198762819	0.33018761905	0.314872483	0.2928942
		RSK	0.3664874785	0.34547018091	0.33015780438	0.311870659	0.2871195
		RDK	0.3672891487	0.34741258453	0.33005416742	0.307816138	0.3007148
	0.7	PDQM	0.3671068214	0.34792114513	0.35267014237	0.330101082	0.2910412
		RSK	0.3741014734	0.35432142199	0.33015871538	0.321540739	0.3008924
		RDK	0.3661240338	0.34591012564	0.30547823176	0.310784533	0.2187318

In addition to Tables 4 and 5, we have introduced more numerical solutions to confirm the reliability and accuracy of the suggested techniques in studying the effect of the generalized Caputo fractional derivatives γ, δ with various values of t on the obtained results, as shown in Figures 2 and 3. These figures also demonstrate that the numerical solutions provided by RDK and RSK agree well with the exact solutions at different values of α and x = 0.33. Figures 2 and 3 show that the obtained solutions increase with increasing time at $v = 1$, $k = 2$ and $v = 2$, $k = 2$, but they are inversely proportional to time at $v = k = 1$ and $v = 2$, $k = 1$ and the value of γ. This refers to the fact that the value of v does not affect the type of relation between the solution and time and vice versa when changing value of k.



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

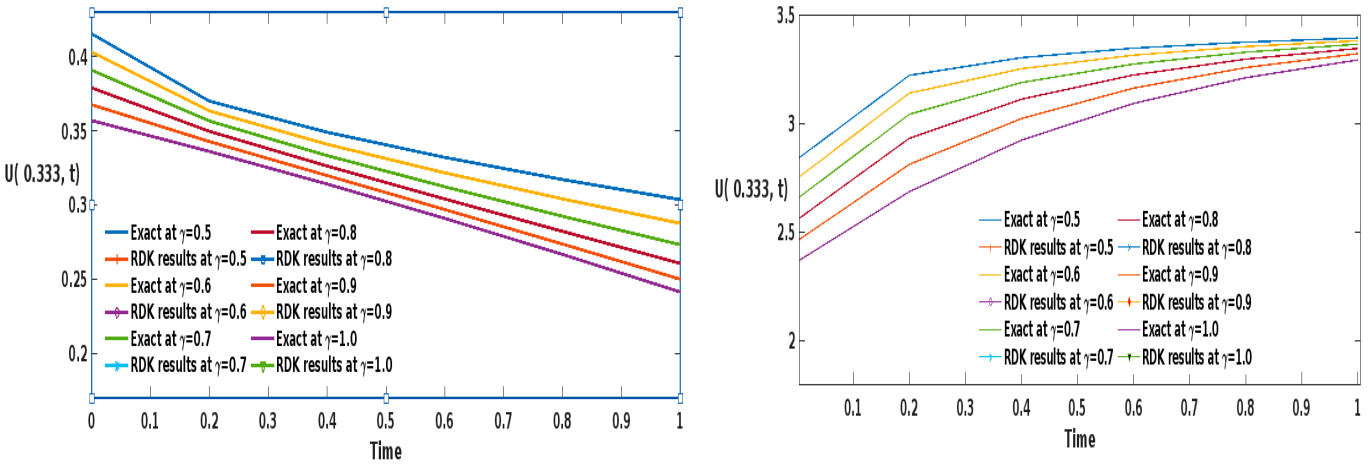
Figure 2. Numerical solutions via RDK and exact solutions at x = 0.333 with different values of time (t) and γ (**a**) k = 1, v = 2. (**b**) k = 2, v = 2.

Furthermore, Figures 4 and 5 explain the influence of parameters ε, v on the obtained solution by RDK at various values of γ, δ in intervals of $1 \leq \varepsilon, v \leq 15$, $0 \leq \alpha \leq 1$. These figures indicate that the obtained solution U is directly proportional to the value of ε, but inversely proportional to v and γ. In Table 6, RMS error is calculated for four schemes

based on different shape functions and combined with the perturbation method at different times ($1 \leq t \leq 6$) s. Thus, the results in this table explain that the value of the RMS error is lowest in the DSCDQ-RDK at all times. Also, this method achieved the lowest computation time. Moreover, in this table, Chebyshev–Gauss–Lobatto nodal points are used to obtain the stable and accurate solution more than uniform PDQM as follows [49]:

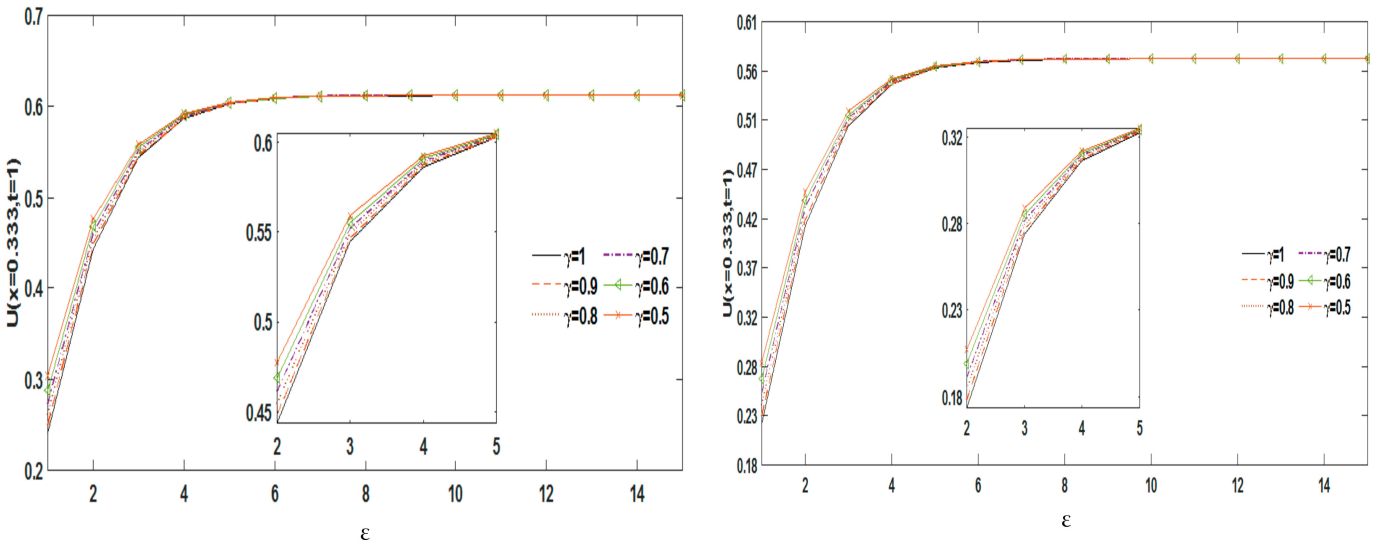
$$X_r = a + \frac{b-a}{2}\left[1 - \cos\frac{(r-1)\pi}{n_x - 1}\right], \quad r = 1, 2, \cdots, n_x, \quad a \leq X \leq b \tag{54}$$

where n_x represents the number of nodal points.



(a) (b)

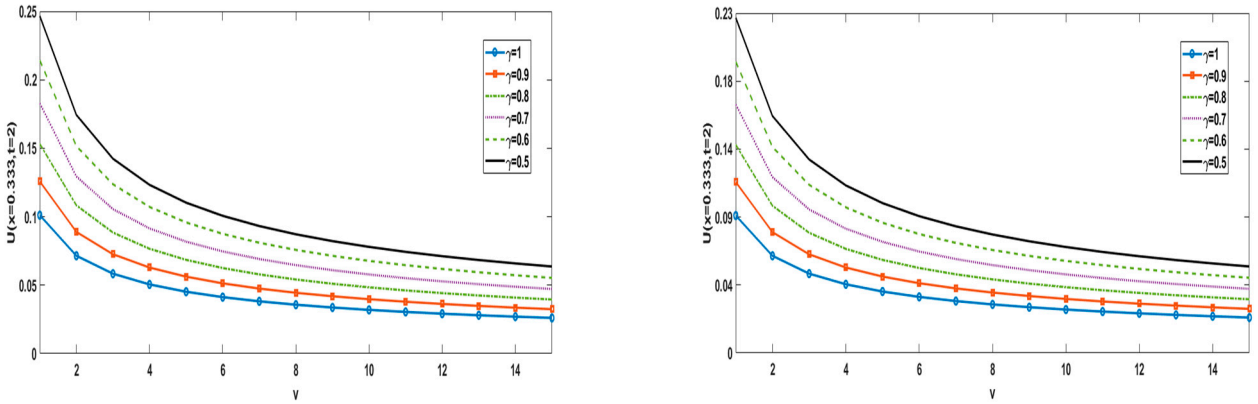
Figure 3. Numerical solutions via RDK and exact solutions at x = 0.333 with different values of time (t) and γ (**a**) v = 1, k = 1. (**b**) v = 1, k = 2.



(a) (b)

Figure 4. Numerical solution by RDK with the parameter ε at various values of γ (**a**) $\delta = 1$ (**b**) $\delta = 0.9$.

Moreover, Figures 6 and 7 show the physical attitude of the obtained numerical solutions for U by RSK and RDK at different values of γ and k by plotting surface graphs of the numerical solutions U. It is remarkable that the obtained numerical solutions decrease with the increasing value of γ and that they are directly proportional to k.

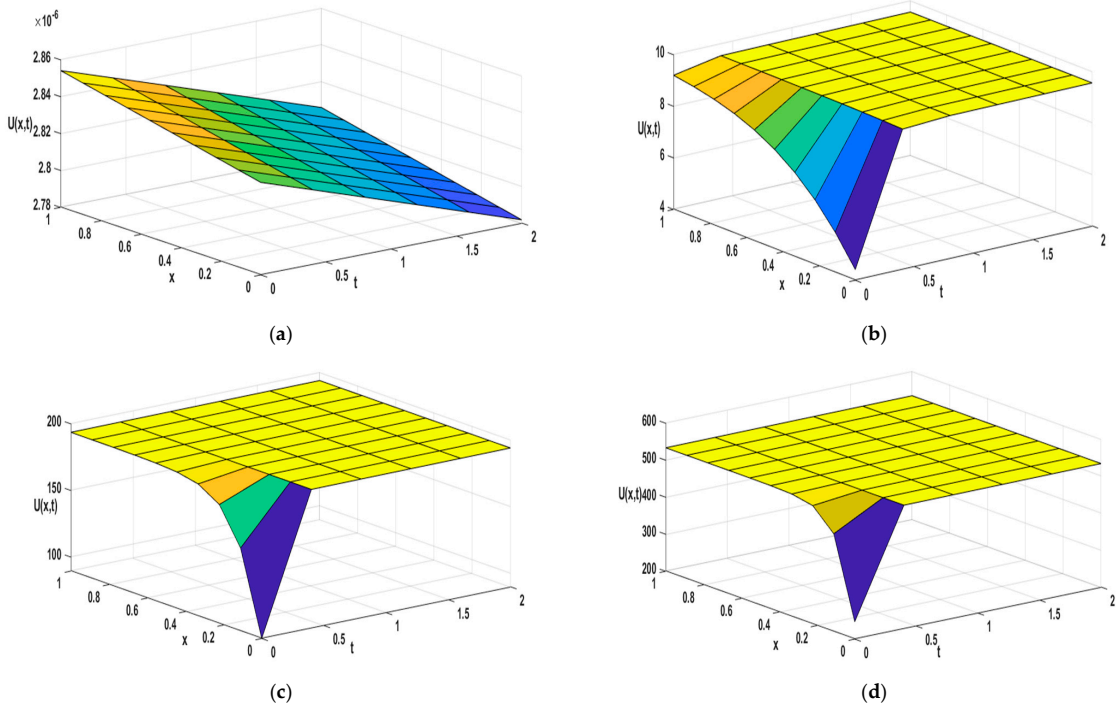


(**a**) $\delta = 1$ (**b**) $\delta = 0.9$

Figure 5. Numerical solution by RDK with the diffusion parameter v at various values of γ (**a**) $\delta = 1$ (**b**) $\delta = 0.9$.

Table 6. The RMS error and computation time for all proposed schemes of DQ when grid size (n_xx $n_t = 5 \times 7$). For the exact solution, $\gamma = 1,\ \delta = 1,\ v = k = \varepsilon = 1, \alpha = (1 \times \Delta)$.

Time	Non-Uniform PDQM		DSCDQM-RSK		Uniform PDQM		DSCDQM-RDK	
	RMS	Comput. Time (S)	RMS	Comput. Time (S)	RMS	Comput. Time (S)	RMS	Comput. Time (S)
1	5.2629×10^{-5}	0.1375	5.625×10^{-5}	0.08399	6.0666×10^{-5}	0.1374	2.1913×10^{-5}	0.08399
2	1.1454×10^{-4}	0.1556	1.224×10^{-4}	0.08495	1.3201×10^{-4}	0.1544	4.7697×10^{-5}	0.08487
3	1.5468×10^{-4}	0.1640	1.653×10^{-4}	0.0944	1.7824×10^{-4}	0.1640	6.4416×10^{-5}	0.0943
4	2.5786×10^{-4}	0.1782	2.755×10^{-4}	0.1153	2.9704×10^{-4}	0.1781	1.0740×10^{-4}	0.1152
5	7.7367×10^{-4}	0.1832	7.251×10^{-4}	0.1173	8.8964×10^{-4}	0.1811	3.2254×10^{-4}	0.1171
6	7.7812×10^{-4}	0.22108	7.551×10^{-4}	0.15603	9.1604×10^{-4}	0.2210	4.7768×10^{-4}	0.15512



(**a**) (**b**) (**c**) (**d**)

Figure 6. RDK solutions at various values of x, t, and $\gamma = 1$, $\delta = 1$. (**a**) $k = 0.01$ (**b**) $k = 2$ (**c**) $k = 10$ (**d**) $k = 15$.

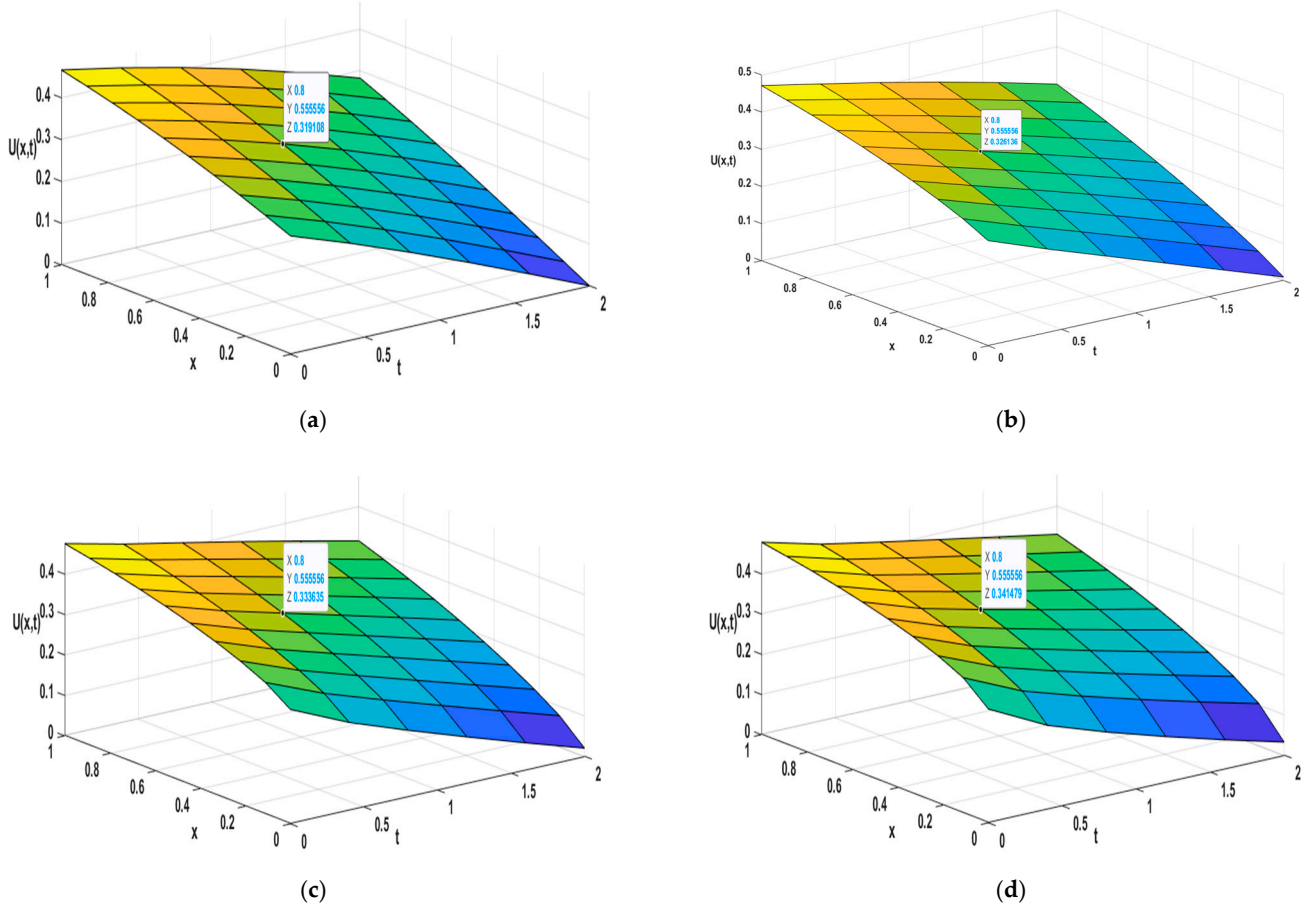


Figure 7. RSK solutions at various values of x, t, and $\gamma, \delta = 1$ (**a**) $\gamma = 1$ (**b**) $\gamma = 0.9$ (**c**) $\gamma = 0.8$ (**d**) $\gamma = 0.7$.

4.2. Consider a Problem Time-Fractional BBM-Burger Equation as Follows

In this problem, we substitute Equation (21) into Equation (5) to transform BBM-Burger equations into nonlinear algebraic equations as follows:

$$\sum_{j=1}^{n_t} \Psi_{tij}^{\gamma,\delta} \mathrm{U}(x,t_j) = \sum_{i=1}^{n_t} \Psi_{tij}^{\gamma,\delta} \mathrm{U}(x,t_j) \sum_{i=1}^{n_x} \Psi_{xx}{}_{ij}^{\gamma,\delta} \mathrm{U}_{(x_i,t)} + \sum_{i=1}^{n_x} \Psi_{x}{}_{ij}^{\gamma,\delta} \mathrm{U}(x_i,t) - \mathrm{U}_{(x_i,t_j)} \sum_{i=1}^{n_x} \Psi_{x}{}_{ij}^{\gamma,\delta} \mathrm{U}(x_i,t) \quad i = (1,n_x)\,,\ j = (1,n_t) \tag{55}$$

The initial and boundary equation for the nonlinear BBM-Burger equation can be taken as follows:

$$\mathrm{U}(\mathrm{x},0) = \mathrm{sec\,h}^2(x/4) \tag{56}$$

$$\mathrm{U}(0,\mathrm{t}) = \mathrm{sec\,h}^2\left(\frac{-\mathrm{t}^\gamma}{3\Gamma(1+\gamma)}\right),\quad \mathrm{U}(1,\mathrm{t}) = \mathrm{sec\,h}^2\left(\frac{1}{4} - \frac{-\mathrm{t}^\gamma}{3\Gamma(1+\gamma)}\right) \tag{57}$$

Also, the exact solution to the BBM-Burger equation can be defined as follows [36]:

$$\mathrm{U}_{\mathrm{exact}}(\mathrm{x},\mathrm{t}) = \mathrm{sec\,h}^2\left[\frac{\mathrm{x}}{4} - \frac{\mathrm{t}^\gamma}{3\Gamma(1+\gamma)}\right],\quad \mathrm{t} > 0,\ \mathrm{x}_1 \le \mathrm{x} \le \mathrm{x}_2 \tag{58}$$

To demonstrate the efficiency and accuracy of the suggested techniques, we calculate $L\infty$ errors at different values of t and γ, as presented in Table 7. The acquired numerical results show that the DSCRDK method is more accurate than RKHSM [36].

Table 7. Error norms in [0, 1] at different times when $n_x \times n_t = 6 \times 6$ and $\delta = 1$, $\alpha = (1 \times \Delta)$.

Time	γ=0.5		γ=0.75		γ=0.95	
	RKHSM [36]	DSC-RDK $M = 1$	RKHSM [36]	DSC-RDK $M = 1$	RKHSM [36]	DSC-RDK $M = 1$
0.1	5.93×10^{-3}	1.11×10^{-4}	4.42×10^{-3}	1.01×10^{-4}	5.67×10^{-3}	1.10×10^{-4}
0.2	4.23×10^{-3}	2.21×10^{-4}	2.97×10^{-3}	2.11×10^{-4}	4.57×10^{-3}	2.01×10^{-4}
0.3	2.66×10^{-3}	2.21×10^{-4}	1.65×10^{-3}	2.20×10^{-4}	3.58×10^{-3}	2.11×10^{-4}
0.4	1.14×10^{-3}	2.35×10^{-4}	4.93×10^{-4}	2.30×10^{-4}	2.68×10^{-3}	2.27×10^{-4}
0.5	4.91×10^{-4}	2.01×10^{-4}	7.94×10^{-4}	2.01×10^{-4}	1.88×10^{-3}	2.00×10^{-4}
0.6	1.96×10^{-3}	2.93×10^{-4}	1.87×10^{-3}	2.21×10^{-4}	1.20×10^{-3}	2.18×10^{-4}
0.7	3.46×10^{-3}	2.15×10^{-4}	2.89×10^{-3}	2.12×10^{-4}	7.23×10^{-3}	2.11×10^{-4}
0.8	4.98×10^{-3}	2.98×10^{-4}	3.87×10^{-3}	2.54×10^{-4}	4.35×10^{-4}	2.43×10^{-4}
0.9	6.50×10^{-3}	2.98×10^{-4}	4.81×10^{-3}	2.55×10^{-4}	5.17×10^{-4}	2.45×10^{-4}

In addition, Figures 8 and 9 establish the 3D surface solutions of the exact and numerical solutions using DSCRSK and PDQM. The obtained results using DSCRSK and PDQM are in good accord with the analytical solutions. Figure 10 shows a comparison between the proposed techniques and the exact solution at t = 0.6. The results are directly proportional to space. DSC-RSK and DSC-RDK methods are more accurate than PDQM. Also, the results are slightly decreased with decreasing δ. Table 8 displays the comparison between the DSC-RSK and DSC-RDK methods. This table shows that the results produced via DSC-RDK are more stable and convergence than DSC-RSK. Also, the presented results are in good agreement with the exact solutions at grid size (5×5). Furthermore, the values of statistical analysis as L_2 errors = 2.7×10^{-5}, while the values for L_∞ errors = 3.2×10^{-7} at (5×5) for the DSC-RSK technique. L_2 errors = 1×10^{-5}, while L_∞ errors =1.2×10^{-7} at (5×5) for the DSC-RDK technique.

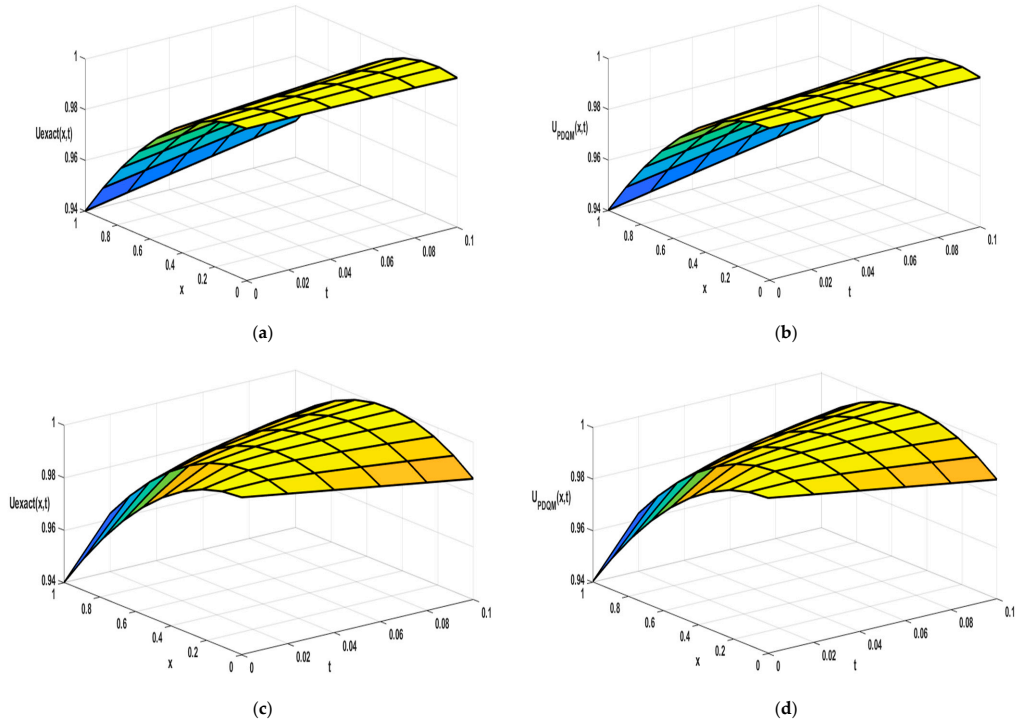


Figure 8. Comparison results when $n_x = 10$, $\delta = 1$, $\gamma = 1, 0.5$. Using (**a**) Exact $\gamma = 1$ (**b**) PDQM $\gamma = 1$ (**c**) Exact $\gamma = 0.5$ (**d**) PDQM $\gamma = 0.5$.

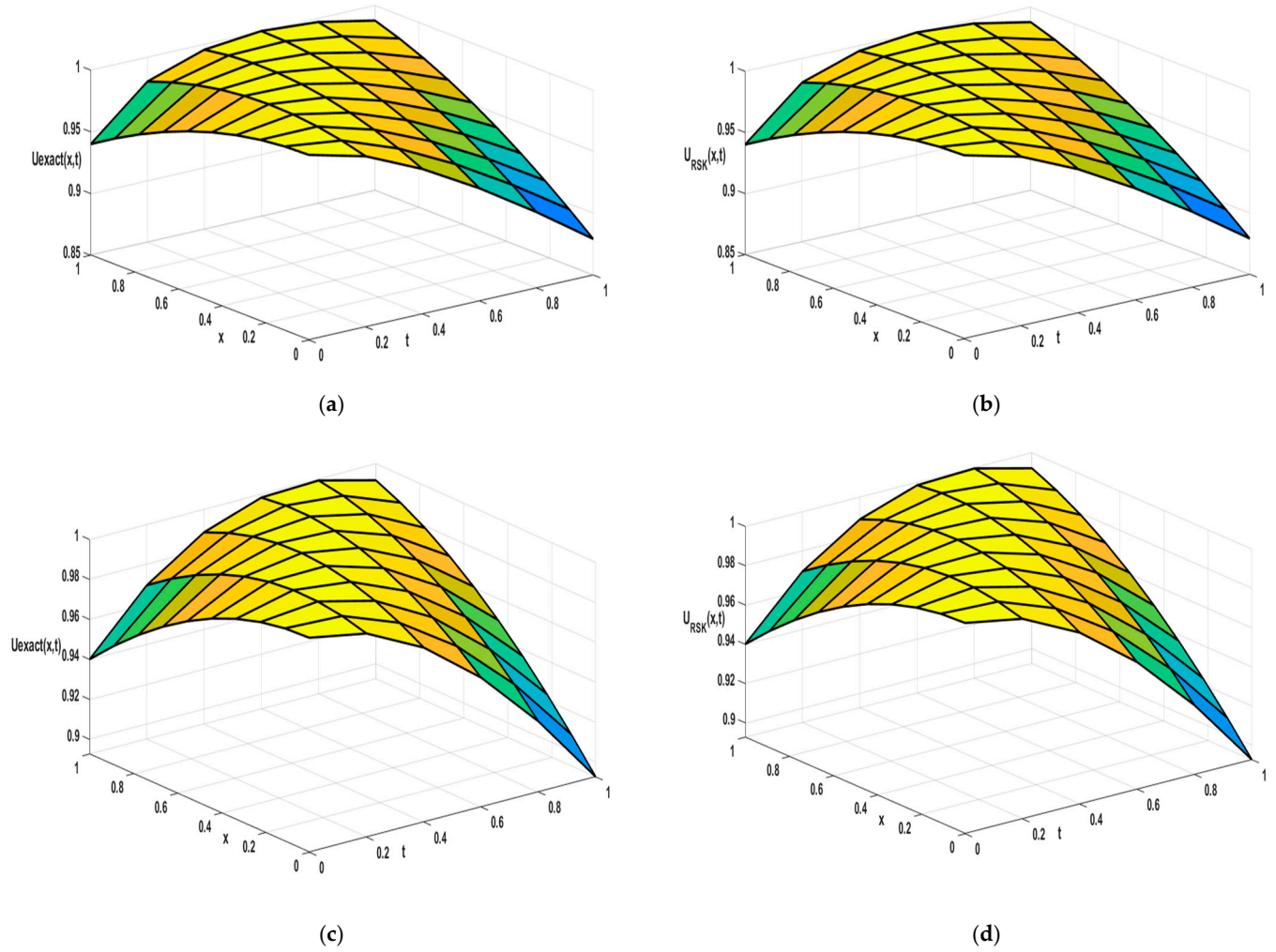


Figure 9. Comparison results when $n_x = 10$, $\delta = 1$, $\gamma = 0.75,\ 0.95$. Using (**a**) exact $\gamma = 0.75$, (**b**) DSCRSK $\gamma = 0.75$, (**c**) exact $\gamma = 0.95$, and (**d**) DSC-RSK $\gamma = 0.95$.

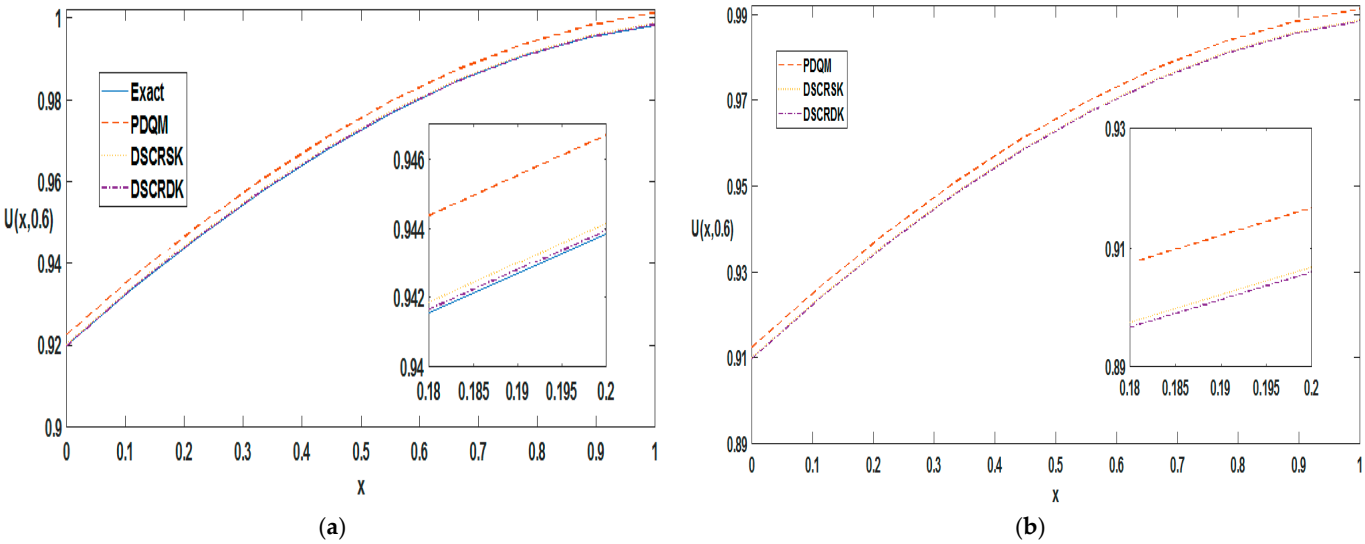


Figure 10. Comparison results of proposed methods with exact solution when $n_x = 10$, $\gamma = 0.5$. (**a**) $\delta = 1$. (**b**) $\delta = 0.7$.

Table 8. L_∞ error norms via DSC-RSK and DSC-RDK at T = 5 s. For the exact solution, $\gamma = 1,\ \delta = 1,\ v = k = \varepsilon = 1, \alpha = (1 \times \Delta)$.

	DSC-RSK		DSC-RDK	
n_x x n_t	L2 Error	L∞ Error	L2 Error	L∞ Error
5×5	2.7987×10^{-5}	3.2532×10^{-7}	1.0557×10^{-5}	1.2922×10^{-7}
5×7	3.2589×10^{-5}	6.3011×10^{-7}	1.4282×10^{-5}	3.2888×10^{-7}
5×8	3.4497×10^{-5}	9.3029×10^{-7}	1.8859×10^{-5}	6.2035×10^{-7}
7×5	3.5214×10^{-5}	9.5135×10^{-7}	1.9551×10^{-5}	6.3141×10^{-7}
7×7	3.7213×10^{-5}	$9.7s141 \times 10^{-7}$	1.9851×10^{-5}	6.5198×10^{-7}

5. Conclusions

Soliton wave solutions for the space–time-fractional modified BBME and BBMBE have been discussed in this paper via novel numerical techniques. These techniques are PDQM, DSCDQM-RSK, and DSCDQM-RDK combined with Caputo type and generalized Caputo's FD. Also, the perturbation and iterative quadrature techniques were used to treat the nonlinearity of this problem. In addition, all results are computed by designing MATLAB code. It is perceived that the obtained numerical solutions are new, more general, and not reported before in the literature. Then, we estimated the study's originality and relevance by comparing its results with those of similar studies. From all the tables and figures presented, it is found that the numerical obtained solutions increase with the increasing parameters v and decreasing ε in problem 1. Also, the results decrease with increasing γ, δ, time, and space.

Accordingly, the obtained numerical solutions show that the suggested techniques are reliable and efficient schemes which yield many complex results for the other fractional NLPDEs. These methods are efficient, powerful, and can be utilized as an alternative method to determine new numerical solutions of several types of fractional DEs applied in mathematical physics.

Author Contributions: Conceptualization, M.M., O.R., M.S. and W.M.A.; methodology, M.M., O.R., M.S. and W.M.A.; software, M.M.; validation, A.M. and W.M.A.; formal analysis, investigation, O.R., M.S. and M.M.; resources, A.M. and W.M.A.; data curation, writing—original draft preparation, O.R., M.S., W.M.A. and M.M.; writing—review and editing, A.M.; visualization, supervision, A.M., O.R. and M.S.; funding, A.M. and W.M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available in the article.

Acknowledgments: The authors would like to acknowledge the use of Gemini 1.5 in the preparation of this manuscript.

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

References

1. Kilbas, A.A.; Srivastava, H.M.; Trujillo, J.J. Theory and applications of fractional differential equations. In *North-Holland Mathematics Studies*; Elsevier Science: Amsterdam, The Netherlands, 2006; Volume 204, pp. 15–23.
2. Miller, K.S.; Ross, B. *An Introduction to the Fractional Calculus and Fractional Differential Equations*; Wiley: New York, NY, USA, 1993.
3. Podlubny, I. *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*; Academic Press: Cambridge, MA, USA, 1998; Volume 198.
4. Gomez-Aguilar, J.F.; Cordova-Fraga, T.; Torres-Jimenez, J.; Escobar-Jimenez, R.F.; Olivares-Peregrino, V.H.; Guerrero-Ramirez, G.V. Nonlocal transport processes and the fractional Cattaneo-Vernotte equation. *Math. Probl. Eng.* **2016**, *2016*, 7845874. [CrossRef]
5. Anjum, N.; He, C.; He, J. Two-Scale Fractal Theory for the Population Dynamics. *Fractals* **2021**, *29*, 2150182. [CrossRef]
6. Abel, N.H. Solutions de quelques problems a l'aide d'integrales definies. *Werk* **1823**, *1*, 10–14.
7. Caputo, M. Linera Models of Dissipation Whose q is Almost Frequency Independent. *Ann. Geophys.* **1966**, *19*, 383–393.

8. Euler, L. On transcendental progression that is, those whose general terms cannot be given algebraically. *Acad. Sci. Imp. Petropolitanae* **1738**, *1*, 38–57.
9. Fourier, J. *Theorie Analytique de la Chaleur*; Gauthier-Villars: Paris, France, 1888; Volume 1.
10. Laplace, P.S. *Theorie Analytique de Probabilities*; Courcier: Paris, France, 1812.
11. Liouville, J. Memory on some questions of geometry and mechanics, and on a new kind of calculation to solve these questions. *J. de l'E'cole Pol. Tech.* **1832**, *13*, 1–69.
12. Ross, B. The development of fractional calculus. 1695–1900. *Hist. Math.* **1977**, *4*, 75–89. [CrossRef]
13. Benjamin, T.B.; Bona, J.L.; Mahony, J.J. Model equations for long waves in nonlinear dispersive systems. *Philos. Trans. R. Soc. A* **1972**, *272*, 47–78.
14. Horton, J.W. Fluctuation spectra of a drift wave soliton gas. *Phys. Fluids* **1982**, *25*, 1838–1843.
15. Islam, S.; Haq, S.; Ali, A. A meshfree method for the numerical solution of the RLW equation. *J. Comput. Appl. Math.* **2009**, *223*, 997–1012. [CrossRef]
16. Kaya, D. A numerical simulation of solitary-wave solutions of the generalized regularized long-wave equation. *Appl. Math. Comput.* **2004**, *149*, 833–841. [CrossRef]
17. Dag, I.; Ozer, M.N. Approximation of RLW equation by least square cubic B-spline finite element method. *Appl. Math. Model.* **2001**, *25*, 221–231. [CrossRef]
18. Dag, I.; Saka, B.; Irk, D. Application of cubic B-splines for numerical solution of the RLW equation. *Appl. Math. Comput.* **2004**, *159*, 373–389.
19. Esen, A.; Kutluay, S. Application of a lumped Galerkin method to the regularized long wave equation. *Appl. Math. Comput.* **2006**, *174*, 833–845. [CrossRef]
20. Gardner, L.R.T.; Gardner, G.A.; Dag, I. A B-spline finite element method for the regularized long wave equation. *Commun. Numer. Methods Eng.* **1995**, *11*, 59–68. [CrossRef]
21. Barati, A. The numerical solution of time fractional generalized Benjamin-Bona-Mahony equation via the Sinc functions. *Int. J. Nonlinear Anal.* **2023**, *14*, 2759–2770.
22. Yaro, D.; Seadawy, A.R.; Dianchen, L.U.; Apeanti, W.O.; Akuamoah, S.W. Dispersive wave solutions of the nonlinear fractional Zakhorov-KuznetsovBenjamin-Bona-Mahony equation and fractional symmetric regularized long wave equation. *Results Phys.* **2019**, *12*, 1971–1979. [CrossRef]
23. Kumar, V. Modified (G′/G)-expansion method for finding traveling wave solutions of the coupled Benjamin–Bona–Mahony-KdV equation. *J. Ocean. Eng. Sci.* **2019**, *4*, 252–255. [CrossRef]
24. Liu, Y. Approximate Solutions of Fractional Nonlinear Equations Using Homotopy Perturbation Transformation Method. In *Abstract and Applied Analysis*; Hindawi Publishing Corporation: Cairo, Egypt, 2012; p. 752869. [CrossRef]
25. Ray, S.S.; Das, G. Numerical simulation of time fractional Benjamin-Bona-Mahony-Burger equation describing propagation of long waves on the water surface. *J. Ocean. Eng. Sci.* **2023**, in press.
26. Dehghan, M.; Shafieeabyaneh, N.; Abbaszadeh, M. Numerical and theoretical discussions for solving nonlinear generalized Benjamin–Bona–Mahony– Burgers equation based on the Legendre spectral element method. *Numer. Methods Partial. Differ. Equ.* **2021**, *37*, 360–382. [CrossRef]
27. Elmandouh, A.; Fadhal, E. Bifurcation of Exact Solutions for the Space-Fractional Stochastic Modified Benjamin–Bona–Mahony Equation. *Fractal Fract.* **2022**, *6*, 718. [CrossRef]
28. Javeeda, S.; Saif, S.; Waheed, A. Dumitru Baleanu, Exact solutions of fractional mBBM equation and coupled system of fractional Boussinesq-Burgers. *Results Phys.* **2018**, *9*, 1275–1281. [CrossRef]
29. Dehghan, M.; Abbaszadeh, M.; Mohebbi, A. The numerical solution of nonlinear high dimensional generalized Benjamin–Bona–Mahony–Burgers equation via the meshless method of radial basis functions. *Comput. Math. Appl.* **2014**, *68*, 212–237. [CrossRef]
30. Oruç, Ö. A new algorithm based on Lucas polynomials for approximate solution of 1D and 2D nonlinear generalized Benjamin–Bona–Mahony–Burgers equation. *Comput. Math. Appl.* **2017**, *74*, 3042–3057. [CrossRef]
31. Dehghan, M.; Abbaszadeh, M.; Mohebbi, A. The use of interpolating element-free Galerkin technique for solving 2D generalized Benjamin–Bona–Mahony–Burgers and regularized long-wave equations on non-rectangular domains with error estimate. *J. Comput. Appl. Math.* **2015**, *286*, 211–231. [CrossRef]
32. Hajiketabi, M.; Abbasbandy, S.; Casas, F. The Lie-group method based on radial basis functions for solving nonlinear high dimensional generalized Benjamin–Bona–Mahony–Burgers equation in arbitrary domains. *Appl. Math. Comput.* **2018**, *321*, 223–243. [CrossRef]
33. Bayarassou, K. Fourth-order accurate difference schemes for solving Benjamin–Bona–Mahony–Burgers (BBMB) equation. *Eng. Comput.* **2019**, *37*, 123–138. [CrossRef]
34. Arora, S.; Jain, R.; Kukreja, V.K. Solution of Benjamin–Bona–Mahony–Burgers equation using collocation method with quintic Hermite splines. *Appl. Numer. Math.* **2020**, *154*, 1–16. [CrossRef]
35. Islam, M.T.; Akbar, M.A.; Azad, M.A. The exact traveling wave solutions to the nonlinear space-time fractional modified Benjamin-Bona-Mahony equation. *J. Mech. Cont. Math. Sci.* **2018**, *13*, 56–71. [CrossRef]
36. Ege, S.M.; Misirli, E. The modified Kudryashov method for solving some fractional-order nonlinear equations. *Adv. Differ. Equ.* **2014**, *2014*, 135. [CrossRef]

37. Guner, O.; Bekir, A. Bright and dark soliton solutions for some nonlinear fractional differential equations. *Chin. Phys. B* **2016**, *25*, 030203. [CrossRef]
38. Kapoor, M.; Joshi, V. A numerical regime for 1-D Burgers' equation using UAT tension B-spline differential quadrature method. *Int. J. Comput. Methods Eng. Sci. Mech.* **2021**, *22*, 181–192. [CrossRef]
39. Joshi, V.; Kapoor, M.; Bhardwaj, N.; Masud, M.; Al-Amri, J.F. Numerical Approximation of One- and Two-Dimensional Coupled Nonlinear Schrödinger Equation by Implementing Barycentric Lagrange Interpolation Polynomial DQM. *Math. Probl. Eng.* **2021**, *2021*, 9968063. [CrossRef]
40. Castro López, R.; Sun, G.-H.; Camacho-Nieto, O.; Yáñez-Márquez, C.; Dong, S.-H. Analytical traveling-wave solutions to a generalized Gross–Pitaevskii equation with some new time and space varying nonlinearity coefficients and external fields. *Phys. Lett. A* **2017**, *381*, 2978–2985. [CrossRef]
41. Guo, Y.; Li, W.; Dong, S.H. Gaussian solitary solution for a class of logarithmic nonlinear Schrödinger equation in (1 + n) dimensions. *Results Phys.* **2023**, *44*, 106187. [CrossRef]
42. Solaimani, M.; Dong, S.-H. Quantum information entropies of multiple quantum well systems in fractional Schrödinger equations. *Int. J. Quant. Chem.* **2020**, *120*, e26113. [CrossRef]
43. Child, M.S.; Dong, S.H.; Wang, X.G. Quantum states of a sextic potential: Hidden symmetry and quantum monodromy. *J. Phys. A* **2000**, *33*, 5653–5661. [CrossRef]
44. Santana-Carrillo, R.; Velázquez Peto, J.M.; Sun, G.-H.; Dong, S.-H. Quantum information entropy for a hyperbolic double well potential in the fractional Schrödinger equation. *Entropy* **2023**, *25*, 988. [CrossRef]
45. Santana-Carrillo, R.; González-Flores, J.S.; Magana-Espinal, E.; Quezada, L.F.; Sun, G.-H.; Dong, S.-H. Quantum information entropy of hyperbolic potentials in fractional schrödinger equation. *Entropy* **2022**, *24*, 1516. [CrossRef]
46. Ragb, O.; Matbuly, M.S.; Civalek, O. Free vibration of irregular plates via indirect differential quadrature and singular convolution techniques. *Eng. Anal. Bound. Elem.* **2021**, *128*, 66–79. [CrossRef]
47. Ragb, O.; Mohamed, M.; Matbuly, M.S.; Civalek, O. An accurate numerical approach for studying perovskite solar cells. *Int. J. Energy Res.* **2021**, *45*, 16456–16477. [CrossRef]
48. Zayed, E.M.E.; Al-Joudi, S. Applications of an extended (G'/G)-expansion method to find exact solutions of nonlinear PDEs in mathematical physics. *Math. Probl. Eng.* **2010**, *2010*, 1–19. [CrossRef]
49. Shu, C. *Differential Quadrature and Its Application in Engineering*; Springer-Verlag: Berlin/Heidelberg, Germany, 2000; Volume 360.
50. Bellman, R.; Kashef, B.; Lee, E.S.; Vasudevan, R. Differential quadrature and splines. *Comput. Math. Appl.* **1975**, *1*, 371–376. [CrossRef]
51. Quan, J.R.; Chang, C.T. New insights in solving distributed system equations by the quadrature method—I. Analysis. *Comput. Chem. Eng.* **1989**, *13*, 779–788. [CrossRef]
52. Quan, J.R.; Chang, C.T. New sightings in involving distributed system equations by the quadrature methods—II. *Comput. Chem. Eng.* **1989**, *13*, 717–724.
53. Shu, C.; Richards, B.E. Application of generalized differential quadrature to solve two-dimensional incompressible Navier-Stokes equations. *Int. J. Numer. Methods Fluids* **1992**, *15*, 791–798. [CrossRef]
54. Wei, G.W. Vibration analysis by discrete singular convolution. *J. Sound Vib.* **2001**, *244*, 535–553. [CrossRef]
55. Wei, G.W.; Zhao, Y.B.; Xiang, Y. Discrete singular convolution and its application to the analysis of plates with internal supports. Part 1: Theory and algorithm. *Int. J. Numer. Methods Eng.* **2002**, *55*, 913–946. [CrossRef]
56. Wang, X.; Xu, S. Free vibration analysis of beams and rectangular plates with free edges by the discrete singular convolution. *J. Sound Vib.* **2010**, *329*, 1780–1792. [CrossRef]
57. Ragb, O.; Wazwaz, A.-M.; Mohamed, M.; Matbuly, M.S.; Salah, M. Fractional differential quadrature techniques for fractional order Cauchy reaction-diffusion equations. *Math. Meth. Appl. Sci.* **2023**, *46*, 10216–10233. [CrossRef]
58. Caputo, M. Linear models of dissipation whose Q is almost frequency independent—II. *Geophys. J. Int.* **1967**, *13*, 529–539. [CrossRef]
59. Weilbeer, M. *Efficient Numerical Methods for Fractional Differential Equations and Their Analytical Background*; Mathematics; Papierflieger: Clausthal-Zellerfeld, Germany, 2005.
60. Mustafa, A.; Ragb, O.; Salah, M.; Salama, R.S.; Mohamed, M. Distinctive Shape Functions of Fractional Differential Quadrature for Solving Two-Dimensional Space Fractional Diffusion Problems. *Fractal Fract.* **2023**, *7*, 668. [CrossRef]
61. Xu, Y.; He, Z.; Agrawal, O.P. Numerical and analytical solutions of new generalized fractional diffusion equation. *Comput. Math. Appl.* **2013**, *66*, 2019–2029. [CrossRef]
62. Kumar, P.; Erturk, V.S.; Kumar, A. A new technique to solve generalized Caputo type fractional differential equations with the example of computer virus model. *J. Math. Ext.* **2021**, *15*, 1–23.
63. Hilfer, R. Applications of Fractional Calculus in Physics. World Scientific Publishing Co., Inc.: River Edge, NJ, USA, 2000.
64. Magin, R.L. *Fractional Calculus in Bioengineering*; Begell House Publishers: Danbury, CT, USA, 2006.
65. Samko, S.G.; Kilbas, A.A.; Marichev, O.I. Fractional Integrals and Derivatives, Theory and Applications. Nikolski, S.M., Ed.; Gordon and Breach Science Publishers: Yverdon, Switzerland, 1993.
66. Nadjafi, J.S.; Ghorbani, A. He's homotopy perturbation method: An effective tool for solving nonlinear integral and integro-differential equations. *Comput. Math. Appl.* **2009**, *2009*, 2379–2390. [CrossRef]

67. Turyilmazoglu, M. Some issues on HPM and HAM method: A convergence scheme. *Math. Comput. Model.* **2011**, *53*, 1929–1936. [CrossRef]
68. Ragb, O.; Seddek, L.F.; Matbuly, M.S. Iterative differential quadrature solutions for Bratu problem. *Comput. Math. Appl.* **2017**, *74*, 249–257. [CrossRef]