

RESEARCH

Open Access



On skewed grid point iterative method for solving 2D hyperbolic telegraph fractional differential equation

Ajmal Ali^{1*} and Norhashidah Hj. Mohd. Ali¹

*Correspondence:

ranaajmalali@gmail.com

¹School of Mathematical Sciences,
Universiti Sains Malaysia, Penang,
Malaysia

Abstract

This paper presents the development of a new iterative method for solving the two-dimensional hyperbolic telegraph fractional differential equation (2D-HTFDE) which is central to the mathematical modeling of transmission line satisfying certain relationship between voltage and current waves in specific distance and time. This equation can be obtained from the classical two-dimensional hyperbolic telegraph partial differential equation by replacing the first and second order time derivatives by the Caputo time fractional derivatives of order 2α and α respectively, with $1/2 < \alpha < 1$. The iterative scheme, called the fractional skewed grid Crank–Nicolson FSkG(C-N), is derived from finite difference approximations discretized on a skewed grid rotated clockwise 45° from the standard grid. The skewed finite difference scheme combined with Crank–Nicolson discretization formula will be shown to be unconditionally stable and convergent by the Fourier analysis. The developed FSkG(C-N) scheme will be compared with the fractional Crank–Nicolson scheme on the standard grid to confirm the effectiveness of the proposed scheme in terms of computational complexities and computing efforts. It will be shown that the new proposed scheme demonstrates more superior capabilities in terms of the number of iterations and CPU timings compared to its counterpart on the standard grid but with the same order of accuracy.

Keywords: Caputo fractional derivative; Standard and skewed grid Crank–Nicolson schemes; Time-fractional hyperbolic telegraph equation

1 Introduction

Fractional calculus is of great importance in mathematical modeling of various phenomena in the field of engineering [1, 2], quantum mechanics [3], hydrology [4], viscoelasticity, [5, 6], bio science [7], control system [8], and other sciences [9–13]. The fractional derivative which simultaneously possesses memory and nonlocal property can describe different nonlinear phenomena more accurately and efficiently in comparison with the integer-order derivative. This makes fractional calculus a powerful tool for modeling the complex dynamical systems [14].

Nowadays, fractional calculus [15] is also being used to represent non-differentiable problems for various types of fractional differential equations involving the local-fractional derivatives. In this regard, Singh et al. [16] recommended the non-differentiable solutions

of the local-fractional wave equation in fractal strings by applying local-fractional homotopy perturbation Laplace transform method (LFHPLTM). The uniqueness and convergence are also proved by the suggested method. Young et al. [17] developed a new Boussinesq-type model in a fractal domain based on the local-fractional derivative. To achieve exact traveling wave solutions obtained from the non-differentiable graph, they convert the local-fractional Boussinesq equation into nonlinear local-fractional ordinary differential equations. In 2018, Kumar et al. [18] proposed a non-differentiable solution for the vehicular fractal traffic flow problem. They solved the problem with the help of the local-fractional homotopy perturbation Sumudu transform scheme, and the results were computationally rigorous for similar kinds of fractional differential equations occurring in natural sciences.

Besides this, variable-order fractional derivatives are also being utilized for numerical solution of several types of fractional differential equations which are actually the extension of the classical fractional calculus. Recently, many researchers have established that the variable-order fractional calculus is more accurate than the constant-order types [19] to describe complex physical models. Hajipour et al. [20] suggested a variable-order fractional (VOF) model for the accurate approximate solution of reaction-diffusion equation in which they utilized a weighted shifted Grunwald derivative to solve the fractional part of the time derivative. This discretization technique was superior when compared with other methods in the literature. The variable-order fractional calculus is an efficient tool to predict the compression deformation of amorphous glassy polymers as well. In 2019, Meng et al. [21] proposed a three-regions-fitting-method and found that VOF models are very efficient with higher accuracy. In [22], Cao et al. derived the excellent results for variable-order time sub-diffusion equation by utilizing Crank–Nicolson type compact finite difference scheme with second-order temporal and fourth-order spatial accuracy.

Mathematical models can be presented more accurately by using fractional calculus than the classical calculus to illustrate the characteristics of the real-world phenomena in an inclusive manner. For example, Jajarmi and Baleanu [23] formulated a perfect model for describing the pathological behavior of HIV infection in fractional calculus that can never investigate the asymptomatic behaviors during the modeling with the integer-order derivatives. In 2018, Baleanu et al. [24] examined their proposed model with regard to the poor nutrition in the life cycle of humans with Mittag-Leffler non-singular kernel and Caputo fractional derivative sense. The comparative numerical analysis reveals that the model based on the fractional derivative with Mittag-Leffler kernel has a different asymptotic behavior than the classic integer-order derivatives. The same type of Mittag-Leffler kernel associated with fractional operator is utilized by Kumar et al. [25] and Baleanu et al. [26] for regularized long-wave equation and nonlinear dynamical system respectively. Shamasneh et al. [27] proposed local fractional entropy (LFE) based model for kidney images enhancement where the quality of MRI images is unpredictable. They tested the model on the poor-quality kidney images and observed the excellent results when LFE techniques were applied. Thus, the new aspects of fractional calculus provide us with more stretchy mathematical models that help us to formulate more realistic judgments about real-world problems in a better way.

In the past few years, a number of numerical or approximation techniques have been developed to solve the hyperbolic telegraph partial differential equation such as homotopy perturbation techniques [28], collocation method [29], reduced differential trans-

form method [30], B-Spline differential quadrature method [31], unconditionally stable ADI methods [32–35], and many other methods [36–38]. However, in solving hyperbolic telegraph fractional differential equation (HTFDE), it is not as straightforward and can be very difficult due to the double time-fractional derivative involved in the equation. Therefore, it is almost impossible for researchers to find the exact analytical solution of such type of fractional differential equation.

Recently, a number of mathematical methods, techniques, and approximations have been proposed in solving hyperbolic telegraph partial differential equation of fractional order due to their importance in representing the various physical problems. Shivanian [39] utilized the spectral meshless radial point interpolation (SMRPI) method for solving two-dimensional fractional telegraph equation. Ferreira et al. [40] solved the multi-dimensional time fractional telegraph equation by using a Fourier transform. They obtained an integral representation of the solution in the Fourier domain expressed in terms of multivariate Mittag-Leffler function which leads to a double Mellin–Barnes type integral representation and consequently to an H-function of two variables. Shivanian et al. [41] used the moving least squares approximation method in which a meshless Galerkin weak form is applied to the interior nodes, whereas the meshless collocation method is used for the nodes on the boundary. Heydari et al. [42] developed an accurate and efficient Legendre wavelet method for numerical solution of the time fractional telegraph equation where they employed both the operational matrices of fractional integration and differentiation to get an accurate numerical solution of the equation. In [43], Hosseini et al. proposed a local radial point interpolant approximation method using radial basis function in solving 2D fractional telegraph equation.

Iterative numerical methods derived from skewed and standard grids have also been researched quite extensively since 1980s in solving various types of partial differential equations (PDEs) including two- and three-dimensional hyperbolic telegraph equations [44–52]. In solving these PDEs, suitably formulated finite difference discretization approximation based on standard grid and skewed grid stencils will generate a large and sparse system of simultaneous linear equations in the form $Au = b$, where A is a known non-singular sparse matrix, b is a constant column vector, and u is the unknown column vector. Due to the sparsity of the matrix A , iterative methods are more suitable in solving the system of linear equations. In all of the cases tested, numerical experimental results show that the iterative schemes derived from the skewed grids exhibited more rapid convergence than the ones derived from the standard grids. This is due to the fact that the skewed finite difference approximations enable us to construct iterative schemes where the iterations need only involve nearly half of the total grid points. The computation of solutions at the remaining grid points will be implemented directly once after the convergence is achieved. This strategy will result in lower arithmetic operations and substantial reduction in the computational complexity of the formulated algorithms.

Because of their good performances in solving integer derivative PDEs, efforts are now being made to apply the strategies on skewed grids in solving fractional differential equations. Preliminary work has been done recently by Balasim and Ali [53–55] on the 2D time fractional diffusion equation, whereby promising results were obtained when the Caputo time fractional was used in combination with implicit skewed approximation formulas for the spatial fractional derivatives. However, the implementation of these skewed approximations on fractional hyperbolic differential equations is still an open question. In this

paper, we have extended the formulation of the skewed grid iterative method on solving the more complicated 2D-HTFDE and observed the efficient numerical results. To the best of our knowledge, this work has not been done by other researchers.

The organization of this paper is as follows: In Sect. 2, we first derive the Caputo time fractional approximations for 2α and α and then the fractional standard grid Crank–Nicolson FSG(C-N) iterative scheme using the standard five point finite difference formula in hybrid with the Crank–Nicolson approximation formula with respect to the standard grid. In Sect. 3, we formulate the fractional skewed grid Crank–Nicolson FSkG(C-N) iterative scheme by rotating the standard grid clockwise to an angle of 45° . The stability and convergence of FSkG(C-N) iterative scheme are discussed in Sects. 4 and 5 respectively. In Sect. 6, a solvability property of the proposed FSkG(C-N) scheme is presented, followed by some numerical experimental results and computational complexities analysis in Sect. 7. The conclusion of the work is presented in Sect. 8.

2 FSG(C-N) scheme for 2D-HTFDE

In this section, we present a brief explanation about the discrete derivation of time-fractional derivative of order 2α and α , where $1/2 < \alpha < 1$, and derive an FSG(C-N) iterative scheme based on the standard five point grid Crank–Nicolson discretizations.

The two-dimensional second-order hyperbolic telegraph time-fractional differential equation is

$$\frac{\partial^{2\alpha} u}{\partial t^{2\alpha}} + 2v(x, y, t) \frac{\partial^\alpha u}{\partial t^\alpha} + \gamma^2(x, y, t)u = a(x, y, t) \frac{\partial^2 u}{\partial x^2} + b(x, y, t) \frac{\partial^2 u}{\partial y^2} + f(x, y, t), \tag{1}$$

where $v(x, y, t) > 0$, $\gamma(x, y, t) > 0$, $a(x, y, t) > 0$, $b(x, y, t) > 0$, and $1/2 < \alpha < 1$, this implies $1 < 2\alpha < 2$ with initial and boundary conditions

$$\begin{aligned} u(x, y, 0) &= \phi(x, y), & u_t(x, y, 0) &= 0, \\ u(0, y, t) &= g_1(y, t), & u(L, y, t) &= g_2(y, t), \\ u(x, 0, t) &= g_3(x, t), & u(x, L, t) &= g_4(x, t), \end{aligned}$$

where $\Omega = \{(x, y, t) / 0 \leq x, y \leq L, 0 \leq t \leq T\}$.

If $v > 0$ and $\gamma = 0$, then Eq. (1) represents the damped wave equation, and if both $v, \gamma > 0$, then it represents the 2D-HTFDE.

The Caputo time fractional derivative is defined as follows:

$$\frac{\partial^\alpha u(x, y, t)}{\partial t^\alpha} = \frac{1}{\Gamma(m - \alpha)} \int_0^t \frac{\partial u^m(x, y, \xi)}{\partial \xi^m} \frac{d\xi}{(t - \xi)^{\alpha+1-m}}, \quad m - 1 < \alpha < m. \tag{2}$$

In this paper, we choose a uniform grid of mesh points with $t_k = k\tau, k = 0, 1, 2, \dots, N, x_i = i\Delta x, i = 0, 1, 2, \dots, M_x$, and $y_j = j\Delta y, j = 0, 1, 2, \dots, M_y$, where N, M_x , and M_y are the positive integers and $\tau = \frac{T}{N}, \Delta x = \frac{L}{M_x}$, and $\Delta y = \frac{L}{M_y}$ are the uniform temporal and spatial mesh sizes respectively. The grid points in the internal grid space are given by $(x_i, y_j, t_k) = (ix, jy, k\tau)$, where $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$. Let $U_{i,j}^k$ be the exact solution and $u_{i,j}^k$ be the approximate solution of the fractional differential equation (1) at the grid point (x_i, y_j, t_k) and consider $f(x_i, y_j, t_k) = f_{i,j}^k, v(x_i, y_j, t_k) = v_{i,j}^k, \gamma(x_i, y_j, t_k) = \gamma_{i,j}^k, a(x_i, y_j, t_k) = a_{i,j}^k$, and $b(x_i, y_j, t_k) = b_{i,j}^k$.

To solve Eq. (1), we utilize the standard five point grid Crank–Nicolson approximation for space and the Caputo fractional derivative of order 2α and α . To derive the 2α -order time-fractional derivative for Eq. (1), we discretize the second-order differential operator as follows:

$$\frac{\partial^2 u(x, y, \xi)}{\partial \xi^2} = \frac{\partial^2 u(x, y, t_{s+1/2})}{\partial \xi^2} = \frac{u_{ij}^{k+1} - 2u_{ij}^k + u_{ij}^{k-1}}{\tau^2} + O(\tau^2), \quad t_s < \xi < t_{s+1}. \tag{3}$$

Using Eq. (3) into Eq. (2), we can find an efficient approximation for fractional derivative of order 2α , where $1/2 < \alpha < 1$, at the point (x_i, y_j, t_k) as follows:

$$\begin{aligned} & \frac{\partial^{2\alpha} u(x_i, y_j, t_k)}{\partial t^{2\alpha}} \\ &= \frac{1}{\Gamma(2-2\alpha)} \int_0^{t_k} \frac{\partial^2 u(x_i, y_j, \xi)}{\partial \xi^2} \cdot \frac{\partial \xi}{(t_k - \xi)^{2\alpha-1}} \\ &= \frac{1}{\Gamma(2-2\alpha)} \sum_{s=1}^k \int_{(s-1)\tau}^{s\tau} \frac{\partial^2 u(x_i, y_j, \xi)}{\partial \xi^2} \cdot \frac{\partial \xi}{(t_k - \xi)^{2\alpha-1}} \\ &= \frac{1}{\Gamma(2-2\alpha)} \sum_{s=1}^k \int_{(s-1)\tau}^{s\tau} \left[\frac{u_{ij}^{s+1} - 2u_{ij}^s + u_{ij}^{s-1}}{\tau^2} + O(\tau^2) \right] \cdot \frac{\partial \xi}{(t_k - \xi)^{2\alpha-1}} \\ &= \frac{1}{\Gamma(2-2\alpha)} \sum_{s=1}^k \left[\frac{u_{ij}^{s+1} - 2u_{ij}^s + u_{ij}^{s-1}}{\tau^2} + O(\tau^2) \right] \cdot \int_{(s-1)\tau}^{s\tau} \frac{1}{\rho^{2\alpha-1}} \cdot \partial \rho \\ &= \frac{\tau^{2-2\alpha}}{\Gamma(3-2\alpha)} \sum_{s=1}^k \left[\frac{u_{ij}^{s+1} - 2u_{ij}^s + u_{ij}^{s-1}}{\tau^2} + O(\tau^2) \right] [(k-s+1)^{2-2\alpha} - (k-s)^{2-2\alpha}] \\ &= \frac{1}{\Gamma(3-2\alpha)\tau^{2\alpha}} \sum_{s=1}^k b_s^* [u_{ij}^{k-s+2} - 2u_{ij}^{k-s+1} + u_{ij}^{k-s}] \\ & \quad + \frac{1}{\Gamma(3-2\alpha)} \sum_{s=1}^k O(\tau^{4-2\alpha}) [(k-s+1)^{2-2\alpha} - (k-s)^{2-2\alpha}], \end{aligned}$$

where $b_s^* = s^{2-2\alpha} - (s-1)^{2-2\alpha}$, $s = 1, 2, \dots, n$, and $\rho = (t_k - \xi)$ or

$$\frac{\partial^{2\alpha} u(x_i, y_j, t_k)}{\partial t^{2\alpha}} = \frac{1}{\Gamma(3-2\alpha)\tau^{2\alpha}} \sum_{s=1}^k b_s^* [u_{ij}^{k-s+2} - 2u_{ij}^{k-s+1} + u_{ij}^{k-s}] + \frac{k^{1-\alpha}}{\Gamma(3-2\alpha)} O(\tau^{4-2\alpha}).$$

The $4-2\alpha$ order approximation for the Caputo fractional derivative is then given by the following equation:

$$\frac{\partial^{2\alpha} u(x_i, y_j, t_k)}{\partial t^{2\alpha}} = \frac{1}{\Gamma(3-2\alpha)\tau^{2\alpha}} \sum_{s=1}^k b_s^* [u_{ij}^{k-s+2} - 2u_{ij}^{k-s+1} + u_{ij}^{k-s}]. \tag{4}$$

To derive the α -order time-fractional derivative for Eq. (1), we consider the first-order differential operator as follows:

$$\frac{\partial u(x_i, y_j, \xi)}{\partial \xi} = \frac{\partial u(x_i, y_j, t_{s+1/2})}{\partial \xi} = \frac{u_{ij}^{k+1} - u_{ij}^k}{\tau} + O(\tau), \quad t_s < \xi < t_{s+1}. \tag{5}$$

Using Eq. (5) into Eq. (2), we can find an efficient approximation for fractional derivative of order α , where $1/2 < \alpha < 1$, as follows:

$$\begin{aligned} \frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial t^\alpha} &= \frac{1}{\Gamma(1-\alpha)} \int_0^{t_k} \frac{\partial u(x_i, y_j, \xi)}{\partial \xi} \cdot \frac{\partial \xi}{(t_k - \xi)^\alpha} \\ &= \frac{1}{\Gamma(1-\alpha)} \sum_{s=1}^k \int_{(s-1)\tau}^{s\tau} \frac{\partial u(x_i, y_j, \xi)}{\partial \xi} \cdot \frac{\partial \xi}{(t_k - \xi)^\alpha} \\ &= \frac{1}{\Gamma(1-\alpha)} \sum_{s=1}^k \int_{(s-1)\tau}^{s\tau} \left[\frac{u_{i,j}^{s+1} - u_{i,j}^s}{\tau} + O(\tau) \right] \cdot \frac{\partial \xi}{(t_k - \xi)^\alpha} \\ &= \frac{1}{\Gamma(1-\alpha)} \sum_{s=1}^k \left[\frac{u_{i,j}^{s+1} - u_{i,j}^s}{\tau} + O(\tau) \right] \cdot \int_{(s-1)\tau}^{s\tau} \frac{1}{\rho^\alpha} \cdot \partial \rho \\ &= \frac{\tau^{1-\alpha}}{\Gamma(2-\alpha)} \sum_{s=1}^k \left[\frac{u_{i,j}^{s+1} - u_{i,j}^s}{\tau} + O(\tau) \right] [(k-s+1)^{1-\alpha} - (k-s)^{1-\alpha}] \\ &= \frac{1}{\Gamma(2-\alpha)\tau^\alpha} \sum_{s=1}^k b_s [u_{i,j}^{k-s+2} - u_{i,j}^{k-s+1}] \\ &\quad + \frac{1}{\Gamma(2-\alpha)} \sum_{s=1}^k O(\tau^{2-\alpha}) [(k-s+1)^{1-\alpha} - (k-s)^{1-\alpha}], \end{aligned}$$

where $b_s = s^{1-\alpha} - (s-1)^{1-\alpha}$, $s = 1, 2, \dots, n$, and $\rho = (t_k - \xi)$ or

$$\frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial t^\alpha} = \frac{1}{\Gamma(2-\alpha)\tau^\alpha} \sum_{s=1}^k b_s [u_{i,j}^{k-s+2} - u_{i,j}^{k-s+1}] + \frac{k^{1-\alpha}}{\Gamma(2-\alpha)} O(\tau^{2-\alpha}).$$

The $2 - \alpha$ order approximation for the Caputo fractional derivative is then given by the following equation:

$$\frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial t^\alpha} = \frac{1}{\Gamma(2-\alpha)\tau^\alpha} \sum_{s=1}^k b_s [u_{i,j}^{k-s+2} - u_{i,j}^{k-s+1}]. \tag{6}$$

Lemma 1 *The coefficients b_s , $s = 0, 1, 2, \dots$, defined in Eqs. (4) and (6) satisfy the following properties:*

- (1). $b_1 = 1, b_s > 0, \forall s = 1, 2, \dots, k$,
- (2). $b_{s-1} > b_s, \forall s = 1, 2, \dots, k$,
- (3). $b_s^{-1} \leq b_k^{-1}, \forall s = 1, 2, \dots, k$,
- (4). $\sum_{s=1}^{k-1} (b_{k-s} - b_{k-s+1}) + b_k = b_1, \forall s = 1, 2, \dots, k$,
- (5). $\sum_{s=2}^{k-1} (b_s - b_{s-1}) + b_1 = b_{k-1}, \forall s = 1, 2, \dots, k$,
- (6). $\sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) + b_1 = \sum_{s=2}^{k-1} b_s^* + b_{k-1}^*, \forall s = 1, 2, \dots, k$.

Utilizing the Crank–Nicolson scheme and the Caputo fractional derivatives at the point $(x_i, y_j, t_{k+1/2})$ from Eqs. (4) and (6) leads Eq. (1) to the following expression:

$$q \sum_{s=1}^k b_s^* [u_{i,j}^{k-s+2} - 2u_{i,j}^{k-s+1} + u_{i,j}^{k-s}] + p_1 \sum_{s=1}^k b_s [u_{i,j}^{k-s+2} - u_{i,j}^{k-s+1}] + p_2 (u_{i,j}^{k+1} + u_{i,j}^k)$$

$$\begin{aligned}
 &= \frac{r_1}{2} [u_{i-1,j}^{k+1} - 2u_{ij}^{k+1} + u_{i+1,j}^{k+1} + u_{i-1,j}^k - 2u_{ij}^k + u_{i+1,j}^k] \\
 &\quad + \frac{r_2}{2} [u_{i,j-1}^{k+1} - 2u_{ij}^{k+1} + u_{i,j+1}^{k+1} + u_{i,j-1}^k - 2u_{ij}^k + u_{i,j+1}^k] + f_{ij}^{k+1/2}
 \end{aligned}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$, where

$$\begin{aligned}
 m_0 &= \tau^{2\alpha} \Gamma(2 - \alpha) \Gamma(3 - 2\alpha), & q &= \Gamma(2 - \alpha) \\
 p_1 &= p_1(i, j, k) = 2\tau^\alpha \Gamma(3 - 2\alpha) v_{ij}^k, & p_2 &= p_2(i, j, k) = m_0 2^{-1} (\gamma_{ij}^k)^2 \\
 r_1 &= r_1(i, j, k) = a_{ij}^k \frac{m_0}{(\Delta x)^2}, & r_2 &= r_2(i, j, k) = b_{ij}^k \frac{m_0}{(\Delta y)^2}.
 \end{aligned}$$

Rewrite the above equation as

$$\begin{aligned}
 u_{ij}^{k+1} &= \frac{1}{(p_1 + p_2 + r_1 + r_2 + q)} \left[\frac{r_1}{2} (u_{i-1,j}^{k+1} + u_{i+1,j}^{k+1}) + \frac{r_2}{2} (u_{i,j-1}^{k+1} + u_{i,j+1}^{k+1}) + \frac{r_1}{2} (u_{i-1,j}^k + u_{i+1,j}^k) \right. \\
 &\quad + \frac{r_2}{2} (u_{i,j-1}^k + u_{i,j+1}^k) + (2q + p_1 - p_2 - r_1 - r_2) u_{ij}^k - q u_{ij}^{k-1} \\
 &\quad \left. - p_1 \sum_{s=2}^k b_s [u_{ij}^{k-s+2} - u_{ij}^{k-s+1}] - q \sum_{s=2}^k b_s^* [u_{ij}^{k-s+2} - 2u_{ij}^{k-s+1} + u_{ij}^{k-s}] + m_0 f_{ij}^{k+1/2} \right]
 \end{aligned}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$.

Another alternative form is

$$\begin{aligned}
 u_{ij}^{k+1} &= \frac{1}{(p_1 + p_2 + r_1 + r_2 + q)} \left[\frac{r_1}{2} (u_{i-1,j}^{k+1} + u_{i+1,j}^{k+1}) + \frac{r_2}{2} (u_{i,j-1}^{k+1} + u_{i,j+1}^{k+1}) + \frac{r_1}{2} (u_{i-1,j}^k + u_{i+1,j}^k) \right. \\
 &\quad + \frac{r_2}{2} (u_{i,j-1}^k + u_{i,j+1}^k) + (2q + p_1 - p_2 - r_1 - r_2) u_{ij}^k + p_1 (b_k u_{ij}^1 - b_2 u_{ij}^k) \\
 &\quad + q (2b_k^* u_{ij}^1 - b_2^* u_{ij}^k - u_{ij}^{k-1}) + p_1 \sum_{s=2}^{k-1} (b_s - b_{s-1}) u_{ij}^{k-s+1} + q \sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) u_{ij}^{k-s+1} \\
 &\quad \left. - q \sum_{s=2}^{k-1} b_s^* u_{ij}^{k-s} + m_0 f_{ij}^{k+1/2} \right] \tag{7}
 \end{aligned}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$ with initial and boundary conditions

$$\begin{aligned}
 u_{ij}^0 &= \phi(x_i, y_j), \\
 u_{0,j}^k &= g_1(y_j, t), & u_{L,j}^k &= g_2(y_j, t), \\
 u_{i,0}^k &= g_3(x_j, t), & u_{i,L}^k &= g_4(x_j, t),
 \end{aligned}$$

$$0 \leq x, y \leq L, 0 \leq t \leq T.$$

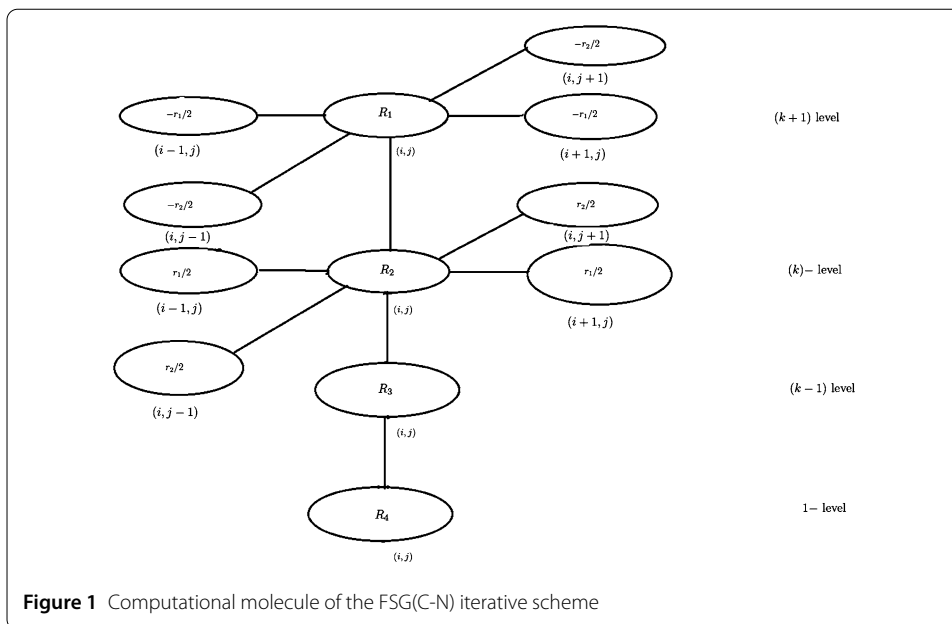


Figure 1 Computational molecule of the FSG(C-N) iterative scheme

As the solution domain is discretized as $u_t(x_i, y_j, 0) = \frac{u^1(x_i, y_j) - u^{-1}(x_i, y_j)}{2\tau}$, from the initial condition we have $u_t(x_i, y_j, 0) = 0$, which implies $u_{i,j}^{-1} = u_{i,j}^1$.

For $k = 0$,

$$u_{i,j}^1 = \frac{1}{(p_1 + p_2 + r_1 + r_2 + 2q)} \left[\frac{r_1}{2} (u_{i-1,j}^1 + u_{i+1,j}^1) + \frac{r_2}{2} (u_{i,j-1}^1 + u_{i,j+1}^1) + \frac{r_1}{2} (u_{i-1,j}^0 + u_{i+1,j}^0) + \frac{r_2}{2} (u_{i,j-1}^0 + u_{i,j+1}^0) + (2q + p_1 - p_2 - r_1 - r_2 - p_1 b_2 - q b_2^*) u_{i,j}^0 + m_0 f_{i,j}^{1/2} \right]$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$.

In Figs. 1 and 2, consider the following:

$$\begin{aligned} R_1 &= p_1 + p_2 + r_1 + r_2 + q, \\ R_2 &= 2q + p_1 - p_2 - r_1 - r_2 - p_1 b_2 - q b_2^*, \\ S_1 &= p_1 + p_2 + r_1/2 + r_2/2 + q, \\ S_2 &= 2q + p_1 - p_2 - r_1/2 - r_2/2 - p_1 b_2 - q b_2^*, \\ R_3 &= p_1(b_2 - 1) + q(2b_2^* - 1) - q, \\ R_4 &= p_1 b_k + 2q b_k^* - q b_{k-1}^*. \end{aligned}$$

The scheme defined in Eq. (7) is an all grid-based numerical scheme on the computational domain which utilizes all the grid points for the grid convergence. In such type of all grid-based numerical schemes, all the grid points of the solution rectangular domain are treated as iterative grid points in the iterative process at each time level until the grid converged solutions are achieved for some predefined convergence criteria and tolerance factor. When grid converged solutions are attained, the estimated grid solutions are then utilized as an initial guess for the next time level.

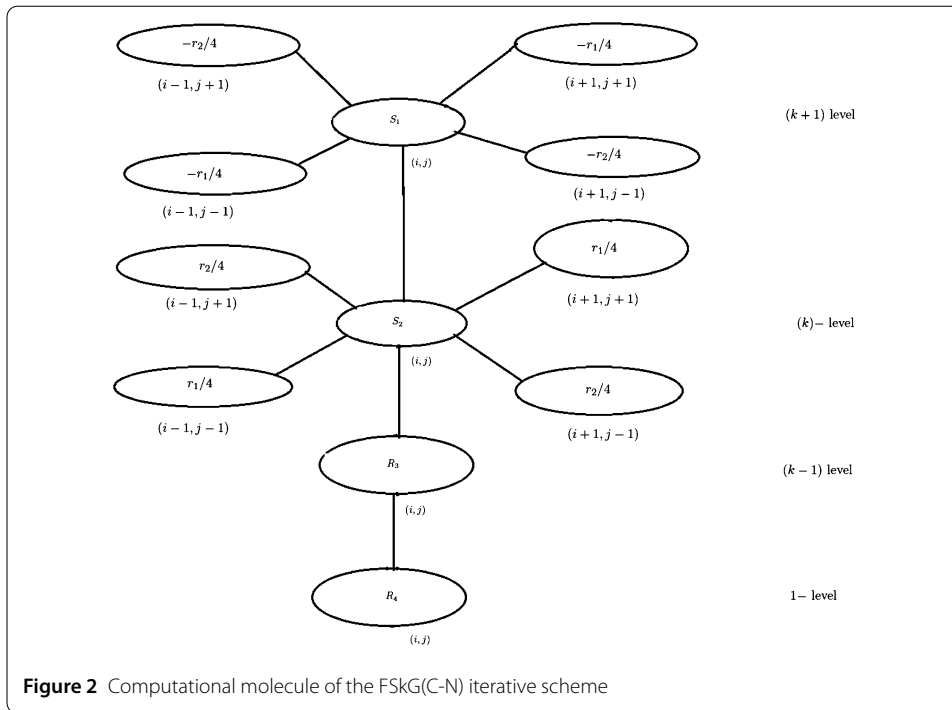


Figure 2 Computational molecule of the FSKG(C-N) iterative scheme

3 FSKG(C-N) scheme for 2D-HTFDE

The expression for an FSKG(C-N) iterative scheme can be derived from the FSG(C-N) iterative scheme by rotating the grids clockwise an angle of 45° with respect to the standard grids. Utilization of skewed Crank–Nicolson difference approximation and the Caputo fractional derivative of order 2α and α from Eqs. (4) and (6) at the point $(x_i, y_j, t_{k+1/2})$ in Eq. (1) leads to the following expression:

$$\begin{aligned}
 & q \sum_{s=1}^k b_s^* [u_{i,j}^{k-s+2} - 2u_{i,j}^{k-s+1} + u_{i,j}^{k-s}] + p_1 \sum_{s=1}^k b_s [u_{i,j}^{k-s+2} - u_{i,j}^{k-s+1}] + p_2 (u_{i,j}^{k+1} + u_{i,j}^k) \\
 & = \frac{r_1}{4} [u_{i-1,j-1}^{k+1} - 2u_{i,j}^{k+1} + u_{i+1,j+1}^{k+1} + u_{i-1,j-1}^k - 2u_{i,j}^k + u_{i+1,j+1}^k] \\
 & \quad + \frac{r_2}{4} [u_{i-1,j+1}^{k+1} - 2u_{i,j}^{k+1} + u_{i+1,j-1}^{k+1} + u_{i-1,j+1}^k - 2u_{i,j}^k + u_{i+1,j-1}^k] + m_0 f_{i,j}^{k+1/2}
 \end{aligned}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$.

Again rewrite the equation as follows:

$$\begin{aligned}
 u_{i,j}^{k+1} = & \frac{1}{(p_1 + p_2 + r_1/2 + r_2/2 + q)} \left[\frac{r_1}{4} (u_{i-1,j+1}^{k+1} + u_{i+1,j-1}^{k+1}) + \frac{r_2}{4} (u_{i-1,j-1}^{k+1} + u_{i+1,j+1}^{k+1}) \right. \\
 & + \frac{r_1}{4} (u_{i-1,j+1}^k + u_{i+1,j-1}^k) + \frac{r_2}{4} (u_{i-1,j-1}^k + u_{i+1,j+1}^k) + (2q + p_1 - p_2 - r_1/2 - r_2/2) u_{i,j}^k \\
 & - q u_{i,j}^{k-1} - p_1 \sum_{s=2}^k b_s [u_{i,j}^{k-s+2} - u_{i,j}^{k-s+1}] \\
 & \left. - q \sum_{s=2}^k b_s^* [u_{i,j}^{k-s+2} - 2u_{i,j}^{k-s+1} + u_{i,j}^{k-s}] + m_0 f_{i,j}^{k+1/2} \right]
 \end{aligned}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$.

An alternative form of the above equation is

$$\begin{aligned}
 u_{ij}^{k+1} = & \frac{1}{(p_1 + p_2 + r_1/2 + r_2/2 + q)} \left[\frac{r_1}{4} (u_{i-1,j+1}^{k+1} + u_{i+1,j-1}^{k+1}) + \frac{r_2}{4} (u_{i-1,j-1}^{k+1} + u_{i+1,j+1}^{k+1}) \right. \\
 & + \frac{r_1}{4} (u_{i-1,j+1}^k + u_{i+1,j-1}^k) + \frac{r_2}{4} (u_{i-1,j-1}^k + u_{i+1,j+1}^k) + (2q + p_1 - p_2 - r_1/2 - r_2/2) u_{ij}^k \\
 & + p_1 (b_k u_{ij}^1 - b_2 u_{ij}^k) + q (2b_k^* u_{ij}^1 - b_2^* u_{ij}^k - u_{ij}^{k-1}) + p_1 \sum_{s=2}^{k-1} (b_s - b_{s-1}) u_{ij}^{k-s+1} \\
 & \left. + q \sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) u_{ij}^{k-s+1} - q \sum_{s=2}^{k-1} b_s^* u_{ij}^{k-s} + m_0 f_{ij}^{k+1/2} \right] \tag{8}
 \end{aligned}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$ with initial and boundary conditions

$$\begin{aligned}
 u_{ij}^0 &= \phi(x_i, y_j), \\
 u_{0,j}^k &= g_1(y_j, t), & u_{L,j}^k &= g_2(y_j, t), \\
 u_{i,0}^k &= g_3(x_j, t), & u_{i,L}^k &= g_4(x_j, t),
 \end{aligned}$$

$0 \leq x, y \leq L, 0 \leq t \leq T$.

For $k = 0$, again from the initial condition, we have

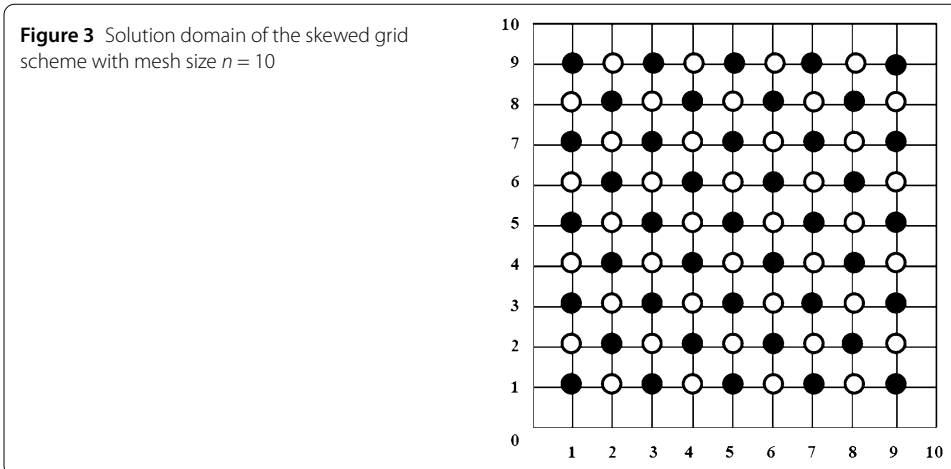
$$\begin{aligned}
 u_{ij}^1 = & \frac{1}{(p_1 + p_2 + r_1/2 + r_2/2 + 2q)} \left[\frac{r_1}{4} (u_{i-1,j+1}^1 + u_{i+1,j-1}^1) + \frac{r_2}{4} (u_{i-1,j-1}^1 + u_{i+1,j+1}^1) \right. \\
 & + \frac{r_1}{4} (u_{i-1,j+1}^0 + u_{i+1,j-1}^0) + \frac{r_2}{4} (u_{i-1,j-1}^0 + u_{i+1,j+1}^0) \\
 & \left. + (2q + p_1 - p_2 - r_1/2 - r_2/2 - p_1 b_2 - q b_2^*) u_{ij}^0 + m_0 f_{ij}^{1/2} \right]
 \end{aligned}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$.

The scheme defined in Eq. (8) is a half-grid based numerical scheme on the solution rectangular domain which utilizes the half of the grid points for the grid convergence. In half-grid based numerical scheme, we divide grid points of the whole discretized domain into sets of two grid points, one type of points is known as iterative points \bullet , while the other type of points is known as direct points \circ as shown in Fig. 3. This process completes in two steps: in the first step the scheme in Eq. (8) iterates only on iterative points \bullet of the solution domain until a certain convergence is achieved. In the second step, when convergence is attained, the values on direct points \circ are evaluated by the scheme defined in Eq. (7). In this way, by sweeping the half of the grid points, we reduce nearly the half of the computational complexity of algorithm which ultimately decreases CPU timings per iteration.

4 Stability analysis of FSKG(C-N) iterative scheme for 2D-HTFDE

In this section, we discuss the stability of the proposed finite difference scheme by the Fourier method [56]. Suppose that ρ_{ij}^k is the approximation solution of Eq. (8) and define



the error

$$\rho_{ij}^k = U_{ij}^k - u_{ij}^k$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$ and

$$\rho^k = [\rho_{1,1}^k, \rho_{1,2}^k, \dots, \rho_{1,M_y-1}^k, \rho_{2,1}^k, \rho_{2,2}^k, \dots, \rho_{2,M_y-1}^k, \rho_{M_x-1,1}^k, \rho_{M_x-1,2}^k, \dots, \rho_{M_x-1,M_y-1}^k]^T.$$

This error satisfies Eq. (8), and we have

$$\begin{aligned} &-\frac{r_1}{4}(\rho_{i-1,j+1}^{k+1} + \rho_{i+1,j-1}^{k+1}) + (p_1 + p_2 + r_1/2 + r_2/2 + q)\rho_{ij}^{k+1} - \frac{r_2}{4}(\rho_{i-1,j-1}^{k+1} + \rho_{i+1,j+1}^{k+1}) \\ &= \frac{r_1}{4}(\rho_{i-1,j+1}^k + \rho_{i+1,j-1}^k) + (2q + p_1 - p_2 - r_1/2 - r_2/2)\rho_{ij}^k + \frac{r_2}{4}(\rho_{i-1,j-1}^k + \rho_{i+1,j+1}^k) \\ &\quad + p_1(b_k \rho_{ij}^1 - b_2 \rho_{ij}^k) + q(2b_k^* \rho_{ij}^1 - b_2^* \rho_{ij}^k - \rho_{ij}^{k-1}) + p_1 \sum_{s=2}^{k-1} (b_s - b_{s-1}) \rho_{ij}^{k-s+1} \\ &\quad + q \sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) \rho_{ij}^{k-s+1} - q \sum_{s=2}^{k-1} b_s^* \rho_{ij}^{k-s} \end{aligned} \tag{9}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$ with initial and boundary conditions

$$\rho_0^k = \rho_L^k = \rho_{ij}^0 = 0.$$

For $k = 0, 1, 2, \dots, L$, define the grid function

$$\rho^k(x, y) = \begin{cases} \rho_{ij}^k, & x_{i-\frac{\Delta x}{2}} < x \leq x_{i+\frac{\Delta x}{2}}, y_{j-\frac{\Delta y}{2}} < y \leq y_{j+\frac{\Delta y}{2}}, \\ 0, & 0 \leq x \leq \frac{\Delta x}{2} \text{ or } L - \frac{\Delta x}{2} \leq x \leq L, \\ 0, & 0 \leq y \leq \frac{\Delta y}{2} \text{ or } L - \frac{\Delta y}{2} \leq y \leq L. \end{cases}$$

We can expand $\rho^k_{(x,y)}$ in Fourier series as

$$\rho^k(x, y) = \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} \lambda^k(l_1, l_2) e^{2\pi\sqrt{-1}(l_1x/L^* + l_2y/L^*)},$$

where

$$\lambda^k(l_1, l_2) = \frac{1}{L^2} \int_0^L \int_0^L \rho^k(x, y) e^{-2\pi\sqrt{-1}(l_1x/L^* + l_2y/L^*)} dx dy.$$

Using the definition of L^2 norm and Parseval's equality, we have

$$\|\rho^k\|_2^2 = \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} \Delta x \Delta y |\rho^k_{ij}|^2 = \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} |\lambda^k(l_1, l_2)|^2.$$

Suppose that the error Eq. (9) has the solution of the following form:

$$\rho^k_{ij} = \lambda^k e^{\sqrt{-1}(\sigma_1 i \Delta x + \sigma_2 j \Delta y)},$$

where $\sigma_1 = 2\pi l_1/L$ and $\sigma_2 = 2\pi l_2/L$.

Substituting the above expression in Eq. (9), we have

$$\begin{aligned} \lambda^{k+1} = & \frac{1}{\mu + p_1 + p_2 + q} \left[\left\{ \mu + p_1(1 - b_2) - p_2 + q(2 - b_2^*) \right\} \lambda^k + (p_1 b_k + 2q b_k^*) \lambda^1 - q \lambda^{k-1} \right. \\ & \left. + p_1 \sum_{s=2}^{k-1} (b_s - b_{s-1}) \lambda^{k-s+1} + q \sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) \lambda^{k-s-1} - q \sum_{s=2}^{k-1} b_s^* \lambda^{k-s} \right], \end{aligned} \tag{10}$$

where $\mu = r_2 \sin^2(\frac{\sigma_1 i \Delta x + \sigma_2 j \Delta y}{2}) + r_1 \sin^2(\frac{\sigma_1 i \Delta x - \sigma_2 j \Delta y}{2})$.

Proposition 1 *The proposed rotated five point finite difference scheme defined in Eq. (8) is unconditionally stable.*

Proof First, we prove the following result with the help of mathematical induction.

If λ^k ($k = 0, 1, 2, \dots, N$) satisfy Eq. (10), then $|\lambda^k| \leq |\lambda^0|$.

For $k = 0$, from Eq. (10) we have

$$|\lambda^1| = \frac{1}{\mu + p_1 + p_2 + 2q} [\mu + p_1(1 - b_2) - p_2 + q(2 - b_2^*)] |\lambda^0|$$

since $\mu > 0$ and coefficients of p_1, p_2 , and q in the numerator are less than the coefficients of p_1 and q in the denominator in the above expression, i.e.,

$$|\lambda^1| \leq |\lambda^0|.$$

Now, assuming that $|\lambda^m| \leq |\lambda^0|$ for $m = 1, 2, \dots, k$, we have

$$|\lambda^{k+1}| \leq \frac{1}{(\mu + p_1 + p_2 + q)} \left[\left\{ \mu + p_1(1 - b_2) - p_2 + q(2 - b_2^*) \right\} |\lambda^k| \right]$$

$$\begin{aligned}
 &+ (p_1 b_k + 2q b_k^*) |\lambda^1| - q |\lambda^{k-1}| \\
 &+ p_1 \sum_{s=2}^{k-1} (b_s - b_{s-1}) |\lambda^{k-s+1}| + q \sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) |\lambda^{k-s-1}| - q \sum_{s=2}^{k-1} b_s^* |\lambda^{k-s}| \Big] \\
 &= \frac{1}{(\mu + p_1 + p_2 + q)} \left[\{ \mu + p_1(1 - b_2) - p_2 + q(2 - b_2^*) \} + (p_1 b_k + 2q b_k^*) \right. \\
 &\quad \left. - q + p_1 \sum_{s=2}^{k-1} (b_s - b_{s-1}) + q \sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) - q \sum_{s=2}^{k-1} b_s^* \right] |\lambda^0|.
 \end{aligned}$$

Utilizing (5) and (6) in Lemma (1) in the above inequality, we get

$$\begin{aligned}
 |\lambda^{k+1}| &\leq \frac{1}{(\mu + p_1 + p_2 + q)} \left[\{ \mu + p_1(1 - b_2) - p_2 + q(2 - b_2^*) \} + (p_1 b_k + 2q b_k^*) \right. \\
 &\quad \left. + p_1(b_{k-1} - b_1) + q(b_{k-1}^* - 2) \right] |\lambda^0|.
 \end{aligned}$$

Collecting the coefficients of p_1 and q in the numerator, we have the following:

$$|\lambda^{k+1}| \leq \frac{1}{(\mu + p_1 + p_2 + q)} \left[\mu + p_1(b_k + b_{k-1} - b_2) - p_2 + q(2b_k^* + b_{k-1}^* - b_2^*) \right] |\lambda^0|.$$

As $\tau \rightarrow 0$, $p_1, p_2, b_k, b_{k-1}, b_k^*$, and b_{k-1}^* all approach to zero, therefore

$$|\lambda^{k+1}| \leq \frac{1}{(\mu + q)} \left[\mu - q(b_2^*) \right] |\lambda^0|.$$

It is observed that $0 < b_2^* < 1$ and $< 0.886 < q < 1$ for all α . Thus, $0 < q b_2^* < 1$, therefore

$$|\lambda^{k+1}| \leq |\lambda^0|.$$

Now, using the above result and Parseval’s equality, we have the following inequality:

$$\|\rho^k\|_2^2 = \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} |\lambda^k(l_1, l_2)|^2 \leq \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} |\lambda^0(l_1, l_2)|^2 = \|\rho^0\|_2^2,$$

that is,

$$\|\rho^k\| \leq \|\rho^0\|,$$

$k = 0, 1, 2, \dots, N$.

Therefore, the proposed scheme defined in Eq. (8) is unconditionally stable. □

5 Convergence of FSKG(C-N) iterative scheme for 2D-HTFDE

In this section, we discuss the convergence of the skewed grid Crank–Nicolson iterative scheme by Fourier analysis. Suppose that $u(x_i, y_j, t_{k+1/2})$ is the exact solution represented by the Taylor series, then the truncation error at $u(x_i, y_j, t_{k+1/2})$ is denoted by $R_{ij}^{k+1/2}$ and defined as follows:

$$R_{ij}^{k+1/2} = q \sum_{s=1}^k b_s^* (u(x_i, y_j, t_{k-s+2}) - 2u(x_i, y_j, t_{k-s+1}) + u(x_i, y_j, t_{k-s}))$$

$$\begin{aligned}
 &+ p_1 \sum_{s=1}^k b_s(u(x_i, y_j, t_{k-s+2}) - u(x_i, y_j, t_{k-s+1})) + p_2((u(x_i, y_j, t_{k+1}) - u(x_i, y_j, t_k)) \\
 &- \frac{m_0 a(x_i, y_j, t_k)}{4(\Delta x)^2} [u(x_{i-1}, y_{j+1}, t_{k+1}) - 2u(x_i, y_j, t_{k+1}) \\
 &+ u(x_{i+1}, y_{j-1}, t_{k+1}) + u(x_{i-1}, y_{j+1}, t_k) - 2u(x_i, y_j, t_k) + u(x_{i+1}, y_{j-1}, t_k)] \\
 &- \frac{m_0 b(x_i, y_j, t_k)}{4(\Delta x)^2} [u(x_{i-1}, y_{j-1}, t_{k+1}) - 2u(x_i, y_j, t_{k+1}) + u(x_{i+1}, y_{j+1}, t_{k+1}) \\
 &+ u(x_{i-1}, y_{j-1}, t_k) - 2u(x_i, y_j, t_k) + u(x_{i+1}, y_{j+1}, t_k)] - m_0 f(x_i, y_j, t_k) \tag{11}
 \end{aligned}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$.

From Eq. (1), we have

$$\begin{aligned}
 R_{ij}^{k+1/2} &= q \sum_{s=1}^k b_s^*(u(x_i, y_j, t_{k-s+2}) - 2u(x_i, y_j, t_{k-s+1}) + u(x_i, y_j, t_{k-s}) - \frac{\partial^{2\alpha} u(x_i, y_j, t_k)}{\partial t^{2\alpha}} \\
 &+ p_1 \sum_{s=1}^k b_s(u(x_i, y_j, t_{k-s+2}) - u(x_i, y_j, t_{k-s+1}) - \frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial t^\alpha} \\
 &+ \frac{m_0 a(x_i, y_j, t_{k+1})}{4(\Delta x)^2} [\frac{\partial^2 u(x_i, y_j, t_k)}{\partial x^2} - \{u(x_{i-1}, y_{j+1}, t_{k+1}) - 2u(x_i, y_j, t_{k+1}) \\
 &+ u(x_{i+1}, y_{j-1}, t_{k+1}) + u(x_{i-1}, y_{j+1}, t_k) - 2u(x_i, y_j, t_k) + u(x_{i+1}, y_{j-1}, t_k)\} \\
 &+ \frac{m_0 b(x_i, y_j, t_{k+1})}{4(\Delta y)^2} [\frac{\partial^2 u(x_i, y_j, t_k)}{\partial y^2} - \{u(x_{i-1}, y_{j-1}, t_{k+1}) - 2u(x_i, y_j, t_{k+1}) \\
 &+ u(x_{i+1}, y_{j+1}, t_{k+1}) + u(x_{i-1}, y_{j-1}, t_k) - 2u(x_i, y_j, t_k) + u(x_{i+1}, y_{j+1}, t_k)\} \\
 &= O(\tau^{4-2\alpha} + \tau^{2-\alpha} + (\Delta x)^2 + (\Delta y)^2).
 \end{aligned}$$

There exists a constant C_1 for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$ such that

$$R_{ij}^{k+1/2} \leq C_1 (\tau^{4-2\alpha} + \tau^{2-\alpha} + (\Delta x)^2 + (\Delta y)^2), \tag{12}$$

where

$$C_1 = \max_{1 \leq i \leq M_x - 1, 1 \leq j \leq M_y - 1, 0 \leq k \leq N} \{C_{ij}^k\}.$$

Define the error function

$$\phi_{ij}^k = u(x_i, y_j, t_k) - u_{ij}^k.$$

From Eq. (11), we have

$$\begin{aligned}
 &- \frac{r_1}{4} (u(x_{i-1}, y_{j+1}, t_{k+1}) + u(x_{i+1}, y_{j-1}, t_{k+1})) + (p_1 + p_2 + r_1/2 + r_2/2 + q)u(x_i, y_j, t_{k+1}) \\
 &- \frac{r_2}{4} (u(x_{i-1}, y_{j-1}, t_{k+1}) + u(x_{i+1}, y_{j+1}, t_{k+1}))
 \end{aligned}$$

$$\begin{aligned}
 &= \frac{r_1}{4} (u(x_{i-1}, y_{j+1}, t_k) + u(x_{i+1}, y_{j-1}, t_k)) + (2q + p_1 - p_2 - r_1/2 - r_2/2)u(x_i, y_j, t_k) \\
 &\quad - \frac{r_2}{4} (u(x_{i-1}, y_{j-1}, t_k) + u(x_{i+1}, y_{j+1}, t_k)) \\
 &\quad - qu(x_i, y_j, t_{k-1}) - q \sum_{s=2}^k b_s^* (u(x_i, y_j, t_{k-s+2}) - 2u(x_i, y_j, t_{k-s+1}) + u(x_i, y_j, t_{k-s})) \\
 &\quad - p_1 \sum_{s=2}^k b_s (u(x_i, y_j, t_{k-s+2}) - u(x_i, y_j, t_{k-s+1})) - m_0 f(x_i, y_j, t_{k+1/2}). \tag{13}
 \end{aligned}$$

To obtain the error equation subtract Eq. (13) from Eq. (8), we have

$$\begin{aligned}
 &-\frac{r_1}{4} (\phi_{i-1,j+1}^{k+1} + \phi_{i+1,j-1}^{k+1}) + (p_1 + p_2 + r_1/2 + r_2/2 + q)\phi_{ij}^{k+1} - \frac{r_2}{4} (\phi_{i-1,j-1}^{k+1} + \phi_{i+1,j+1}^{k+1}) \\
 &= \frac{r_1}{4} (\phi_{i-1,j+1}^k + \phi_{i+1,j-1}^k) + (2q + p_1 - p_2 - r_1/2 - r_2/2)\phi_{ij}^k + \frac{r_2}{4} (\phi_{i-1,j-1}^k + \phi_{i+1,j+1}^k) \\
 &\quad - q\phi_{ij}^{k-1} - q \sum_{s=2}^{k-1} b_s^* (\phi_{ij}^{k-s+2} - 2\phi_{ij}^{k-s+1} + \phi_{ij}^{k-s}) + p_1 \sum_{s=2}^{k-1} b_s^* (\phi_{ij}^{k-s+2} - \phi_{ij}^{k-s+1}) \\
 &\quad + m_0 R_{ij}^{k+1/2}
 \end{aligned}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$.

Its equivalent form is

$$\begin{aligned}
 &-\frac{r_1}{4} (\phi_{i-1,j+1}^{k+1} + \phi_{i+1,j-1}^{k+1}) + (p_1 + p_2 + r_1/2 + r_2/2 + q)\phi_{ij}^{k+1} - \frac{r_2}{4} (\phi_{i-1,j-1}^{k+1} + \phi_{i+1,j+1}^{k+1}) \\
 &= \frac{r_1}{4} (\phi_{i-1,j+1}^k + \phi_{i+1,j-1}^k) + (2q + p_1 - p_2 - r_1/2 - r_2/2)\phi_{ij}^k + \frac{r_2}{4} (\phi_{i-1,j-1}^k + \phi_{i+1,j+1}^k) \\
 &\quad + p_1 (b_k \phi_{ij}^1 - b_2 \phi_{ij}^k) + q(2b_k^* \phi_{ij}^1 - b_2^* \phi_{ij}^k - \phi_{ij}^{k-1}) + p_1 \sum_{s=2}^{k-1} (b_s - b_{s-1}) \phi_{ij}^{k-s+1} \\
 &\quad + q \sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) \phi_{ij}^{k-s+1} - q \sum_{s=2}^{k-1} b_s^* \phi_{ij}^{k-s} + m_0 R_{ij}^{k+1/2} \tag{14}
 \end{aligned}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N - 1$.

This equation satisfies the boundary conditions

$$\begin{aligned}
 \phi_{i,0}^k &= \phi_{i,m}^k = 0, & 1 \leq i \leq M_x - 1, 0 \leq k \leq N - 1, \\
 \phi_{0,j}^k &= \phi_{n,j}^k = 0, & 1 \leq j \leq M_y - 1, 0 \leq k \leq N - 1
 \end{aligned}$$

and the initial conditions

$$\phi_{ij}^0 = 0, \quad 0 \leq i \leq M_x, 0 \leq j \leq M_y.$$

For $k = 0, 1, 2, \dots, N$, define the grid function as

$$\phi^k(x, y) = \begin{cases} \phi_{ij}^k, & x_{i-\frac{\Delta x}{2}} < x \leq x_{i+\frac{\Delta x}{2}}, y_{j-\frac{\Delta y}{2}} < y \leq y_{j+\frac{\Delta y}{2}}, \\ 0, & 0 \leq x \leq \frac{\Delta x}{2} \text{ or } L - \frac{\Delta x}{2} \leq x \leq L, \\ 0, & 0 \leq y \leq \frac{\Delta y}{2} \text{ or } L - \frac{\Delta y}{2} \leq y \leq L, \end{cases}$$

and

$$R^k(x, y) = \begin{cases} R_{ij}^k, & x_{i-\frac{\Delta x}{2}} < x \leq x_{i+\frac{\Delta x}{2}}, y_{j-\frac{\Delta y}{2}} < y \leq y_{j+\frac{\Delta y}{2}}, \\ 0, & 0 \leq x \leq \frac{\Delta x}{2} \text{ or } L - \frac{\Delta x}{2} \leq x \leq L, \\ 0, & 0 \leq y \leq \frac{\Delta y}{2} \text{ or } L - \frac{\Delta y}{2} \leq y \leq L, \end{cases}$$

for $i = 1, 2, 3, \dots, M_x - 1, j = 1, 2, 3, \dots, M_y - 1$, and $k = 0, 1, 2, 3, \dots, N$.

The grid functions $\phi^k(x, y)$ and $R^k(x, y)$ can be expanded in Fourier series:

$$\phi^k(x, y) = \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} \eta^k(l_1, l_2) e^{2\pi\sqrt{-1}(l_1x/L+l_2y/L)},$$

$$R^k(x, y) = \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} \xi^k(l_1, l_2) e^{2\pi\sqrt{-1}(l_1x/L+l_2y/L)},$$

where

$$\eta^k(l_1, l_2) = \frac{1}{L^2} \int_0^L \int_0^L \phi^k(x, y) e^{-2\pi\sqrt{-1}(l_1x/L+l_2y/L)} dx dy, \tag{15}$$

$$\xi^k(l_1, l_2) = \frac{1}{L^2} \int_0^L \int_0^L R^k(x, y) e^{-2\pi\sqrt{-1}(l_1x/L+l_2y/L)} dx dy. \tag{16}$$

Using the definition of l^2 norm and Parseval’s equality, we have

$$\|\phi^k\|_2^2 = \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} \Delta x \Delta y |\phi_{ij}^k|^2 = \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} |\eta^k(l_1, l_2)|^2, \tag{17}$$

$$\|R^k\|_2^2 = \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} \Delta x \Delta y |R_{ij}^k|^2 = \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} |\xi^k(l_1, l_2)|^2, \tag{18}$$

where

$$\phi^k = [\phi_{1,1}^k, \phi_{1,2}^k, \dots, \phi_{1,M_y-1}^k, \phi_{2,1}^k, \phi_{2,2}^k, \dots, \phi_{2,M_y-1}^k, \phi_{M_x-1,1}^k, \phi_{M_x-1,2}^k, \dots, \phi_{M_x-1,M_y-1}^k]^T,$$

$$R^k = [R_{1,1}^k, R_{1,2}^k, \dots, R_{1,M_y-1}^k, R_{2,1}^k, R_{2,2}^k, \dots, R_{2,M_y-1}^k, R_{M_x-1,1}^k, R_{M_x-1,2}^k, \dots, R_{M_x-1,M_y-1}^k]^T.$$

Suppose that the solution has the following forms:

$$\phi_{ij}^k = \eta^k e^{\sqrt{-1}(\sigma_1 i \Delta x + \sigma_2 j \Delta y)}, \tag{19}$$

$$R_{ij}^k = \xi^k e^{\sqrt{-1}(\sigma_1 i \Delta x + \sigma_2 j \Delta y)}, \tag{20}$$

where $\sigma_1 = 2\pi l_1/L$ and $\sigma_2 = 2\pi l_2/L$.

Substituting Eq. (19) and Eq. (20) in equation Eq. (14), we have

$$\eta^{k+1} = \frac{1}{\mu + p_1 + p_2 + q} \left[\{\mu + p_1(1 - b_2) - p_2 + q(2 - b_2^*)\} \eta^k + (p_1 b_k + 2q b_k^*) \eta^1 - q \eta^{k-1} \right]$$

$$\begin{aligned}
 &+ p_1 \sum_{s=2}^{k-1} (b_s - b_{s-1}) \eta^{k-s+1} + q \sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) \eta^{k-s-1} - q \sum_{s=2}^{k-1} b_s^* \eta^{k-s} \\
 &+ m_0 \xi^{k+1/2} \Big], \tag{21}
 \end{aligned}$$

where μ is defined in Sect. 4.

Proposition 2 *Assuming that η^k ($k = 0, 1, 2, \dots, N$) is the solution of Eq. (21), then there exists a positive constant C_0 , so that $|\eta^k| \leq C_0 |\xi^{1/2}|$, $k = 0, 1, 2, \dots, N$.*

Proof From Eq. (15), noticing that $\phi^0 = 0$, we have $\eta^0(l_1, l_2) = 0$. For convergence criteria of the series, there exists a positive constant C_0 of the right-hand side of Eq. (18) such that

$$|\xi^k| = |\xi^k(l_1, l_2)| \leq C^k |\xi^{1/2}(l_1, l_2)| = C_0 |\xi^{1/2}|, \quad k = 0, 1, 2, \dots, N, \tag{22}$$

where $C_0 = \max_{0 \leq k \leq N} \{C^k\}$.

We prove the proposition by mathematical induction.

For $k = 0$ from Eq. (21), we have

$$\eta^1 = \frac{m_0}{(\mu + p_1 + p_2 + 2q)} \xi^{1/2}.$$

Since $\mu \geq 0$, and for all values of $\tau > 0$ and α , we have $0 < m_0 < 1$ and $m_0 < (\mu + p_1 + p_2 + 2q)$.

From Eq. (22), we have

$$|\eta^1| \leq C_0 |\xi^{1/2}|.$$

Assume that $|\eta^m| \leq C_0 |\xi^{1/2}|$, $m = 1, 2, \dots, k$. We show that it is true for $n = k + 1$. Again from Eq. (21) we have

$$\begin{aligned}
 |\eta^{k+1}| &\leq \frac{1}{\mu + p_1 + p_2 + q} \\
 &\times \left[\{ \mu + p_1(1 - b_2) - p_2 + q(2 - b_2^*) \} |\eta^k| + (p_1 b_k + 2q b_k^*) |\eta^1| - q |\eta^{k-1}| \right. \\
 &+ p_1 \sum_{s=2}^{k-1} (b_s - b_{s-1}) |\eta^{k-s+1}| + q \sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) |\eta^{k-s-1}| \\
 &\left. - q \sum_{s=2}^{k-1} b_s^* |\eta^{k-s}| + m_0 \xi^k \right] \\
 &= \frac{1}{\mu + p_1 + p_2 + q} \left[\{ \mu + p_1(1 - b_2) - p_2 + q(2 - b_2^*) \} + (p_1 b_k + 2q b_k^*) \right. \\
 &\left. - q + p_1 \sum_{s=2}^{k-1} (b_s - b_{s-1}) + q \sum_{s=2}^{k-1} (2b_s^* - b_{s-1}^*) - q \sum_{s=2}^{k-1} b_s^* + m_0 \right] C_0 |\xi^{1/2}|.
 \end{aligned}$$

By utilizing (5) and (6) in Lemma (1), the following inequality is obtained:

$$\begin{aligned}
 |\eta^{k+1}| &\leq \frac{1}{\mu + p_1 + p_2 + q} [\{\mu + p_1(1 - b_2) - p_2 + q(2 - b_2^*)\} + (p_1 b_k + 2q b_k^*) \\
 &\quad + p_1(b_{k-1} - b_1) + q(b_{k-1}^* - 2) + m_0] C_0 |\xi^{1/2}| \\
 &= \frac{1}{(\mu + p_1 + p_2 + q)} [\mu + p_1(b_k + b_{k-1} - b_2) - p_2 \\
 &\quad + q(2b_k^* + b_{k-1}^* - b_2^*) + m_0] C_0 |\xi^{1/2}|.
 \end{aligned}$$

Since $m_0 = \tau^{2\alpha} \Gamma(2 - \alpha) \Gamma(3 - 2\alpha)$, $p_1 = 2\tau^\alpha \Gamma(3 - 2\alpha) v_{ij}^k$ and $p_2 = m_0 2^{-1} (\gamma_{ij}^k)^2$ as $\tau \rightarrow 0$ implies that $m_0, p_1, p_2 \rightarrow 0$. Further, as $0 < b_2 < 1$ is a constant positive term, whereas $b_k, b_{k-1}, b_k^*,$ and b_{k-1}^* in the above expression depend on the time level k . From Lemma 1, (2) we have $1 = b_0 > b_1 > b_2 > \dots > b_{k-1} > b_k$ and $1 = b_0^* > b_1^* > b_2^* > \dots > b_{k-1}^* > b_k^*$ as the number of time level k increases (in other words $\tau \rightarrow 0$); consequently, $b_k, b_{k-1}, b_k^*,$ and b_{k-1}^* approach to zero. Therefore, we can write the above inequality as

$$|\eta^{k+1}| \leq \frac{1}{(\mu + q)} [\mu - q(b_2^*)] C_0 |\xi^{1/2}|.$$

As $0 < qb_2^* < 1$. Thus,

$$|\eta^{k+1}| \leq C_0 |\xi^{1/2}|. \quad \square$$

Theorem 1 *The proposed rotated five point finite difference scheme defined in Eq. (8) is l_2 convergent, and the order of convergence is $\tau^{4-2\alpha} + \tau^{2-\alpha} + (\Delta x)^2 + (\Delta y)^2$.*

Proof Applying Proposition (2) and Eqs. (17) and (18) for $k = 0, 1, 2, 3, \dots, N$, we have

$$\|\phi^k\|_2^2 = \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} |\eta^k(l_1, l_2)|^2 \leq \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} C_0^2 |\xi^{1/2}(l_1, l_2)|^2 = C_0^2 \|R^{1/2}\|_2^2.$$

There exists positive constant C_1 as in inequality Eq. (12) such that

$$\begin{aligned}
 \|\phi^k\|_2 &\leq C_0 \|R^{1/2}\|_2 \leq C_0 C_1 (\tau^{4-2\alpha} + \tau^{2-\alpha} + (\Delta x)^2 + (\Delta y)^2), \\
 \|\phi^k\|_2 &\leq C (\tau^{4-2\alpha} + \tau^{2-\alpha} + (\Delta x)^2 + (\Delta y)^2)
 \end{aligned}$$

for $k = 0, 1, 2, 3, \dots, N$, where $C = C_0 C_1$.

This completes the theorem. □

6 Solvability of FSKG(C-N) iterative scheme

The FSKG(C-N) iterative scheme can be written in matrix form as follows:

$$\begin{cases}
 AU^1 = BU^0 + \varphi^0, & n = 0, \\
 AU^{n+1} = BU^n + (p_1b_n + 2qb_n^*)U^1 + qU^{n-1} \\
 \quad + p_1 \sum_{m=2}^{n-1} (b_m - b_{m-1})U^{n-m+1} + q \sum_{m=2}^{n-1} (2b_m^* - b_{m-1}^*)U^{n-m+1} \\
 \quad - q \sum_{l=2}^{n-1} b_m U^{n-m} + \varphi^0, & n \geq 1, \\
 U_{ij}^0 = \phi(x_i, y_j), & 1 \leq i \leq M_x - 1, 0 \leq j \leq M_y, \\
 U_{0,j}^n = g_1^n(0, y_j), & 1 \leq j \leq M_y - 1, 0 \leq n \leq N, \\
 U_{L,j}^n = g_2^n(L, y_j), & 1 \leq j \leq M_y - 1, 0 \leq n \leq N, \\
 U_{i,0}^n = g_3^n(x_i, 0), & 1 \leq i \leq M_x - 1, 0 \leq n \leq N, \\
 U_{i,L}^n = g_4^n(x_i, L), & 1 \leq i \leq M_x - 1, 0 \leq n \leq N,
 \end{cases} \tag{23}$$

where $\varphi^n = [\varphi_0^n, \varphi_1^n, \varphi_2^n, \dots, \varphi_N^n]^T$ and $\varphi_{ij}^{k+1/2} = f(x_i, y_j, t_{n+1/2})$ for $1 \leq i \leq M_x - 1, 1 \leq j \leq M_y - 1, 0 \leq k \leq N$. Consider $r = p_1 + p_2 + r_1/2 + r_2/2 + q$ and $p = 2q + p_1 - p_2 - r_1/2 - r_2/2$ in Eq. (8), the matrices A and B are the pentadiagonal matrices of the following forms:

$$A = \begin{pmatrix}
 r & \dots & -r_1/4 & \dots & -r_2/4 & \dots & 0 \\
 \vdots & r & \dots & -r_1/4 & \dots & -r_2/4 & \vdots \\
 -r_1/4 & \ddots & r & \ddots & \ddots & & \ddots \\
 \vdots & \ddots & \ddots & \ddots & & \ddots & -r_2/4 \\
 -r_2/4 & & -r_1/4 & \ddots & \ddots & & \ddots \\
 \vdots & -r_2/4 & \ddots & \ddots & & \ddots & -r_1/4 \\
 & & \ddots & \ddots & & \ddots & \vdots \\
 0 & \dots & & -r_2/4 & \dots & -r_1/4 & \dots & r
 \end{pmatrix},$$

$$B = \begin{pmatrix}
 p & \dots & r_1/4 & \dots & r_2/4 & \dots & 0 \\
 \vdots & p & \dots & r_1/4 & \dots & r_2/4 & \vdots \\
 r_1/4 & \ddots & p & \ddots & \ddots & & \ddots \\
 \vdots & \ddots & \ddots & \ddots & & \ddots & r_2/4 \\
 r_2/4 & & r_1/4 & \ddots & \ddots & & \ddots \\
 \vdots & r_2/4 & \ddots & \ddots & & \ddots & r_1/4 \\
 & & \ddots & \ddots & & \ddots & \vdots \\
 0 & \dots & & r_2/4 & \dots & r_1/4 & \dots & p
 \end{pmatrix}.$$

Theorem 2 *The difference equation defined by Eq. (23) is uniquely solvable.*

Proof Since $r > 0$, the coefficient matrix defined in Eq. (23) is strictly diagonally dominant matrix. Therefore, matrix A is non-singular matrix. This completes the proof. \square

Table 1 Computational complexity analysis for *FSG(C-N)* and *FSkG(C-N)* methods before convergence

Method	Per iteration	
	(+/-)	(x/÷)
FSG(C-N)	$(25 + 22(k - 1))\lambda^2$	$(17 + (k - 1))\lambda^2$
FSkG(C-N)	$(12.5 + 11(k - 1))(\lambda^2 + 1)$	$(8.5 + 0.5(k - 1))(\lambda^2 + 1)$

Table 2 Computational complexity analysis for *FSG(C-N)* and *FSkG(C-N)* methods after convergence

Method	After convergence	
	(+/-)	(x/÷)
FSG(C-N)	–	–
FSkG(C-N)	$(12.5 + 11(k - 1))(\lambda^2 - 1)$	$(8.5 + 0.5(k - 1))(\lambda^2 - 1)$

Table 3 Total number of arithmetic operations of *FSG(C-N)* and *FSkG(C-N)* methods

Methods	Total operations
FSG(C-N)	$(42 + 23(k - 1))\lambda^2 * lte.$
FSkG(C-N)	$(21 + 11.5(k - 1))(\lambda^2 + 1) * lte. + (21 + 11.5(k - 1))(\lambda^2 - 1)$

7 Numerical experiments and results

In order to show the effectiveness and applicability of FSkG(C-N) method in solving the two-dimensional time-fractional hyperbolic telegraph equation, the numerical experiments were carried out on a PC with Core 2 Duo 2.8 GHz, 2GB of RAM with Window XP SP3 operating system using Cygwin C and Mathematica 11 software. For convenience, the Gauss–Seidel method with relaxation factor ω_e equal to 1 was used throughout our numerical calculation in both examples. For the convergence criteria, the l_∞ norm was used with tolerance factor $\varepsilon = 10^{-5}$.

The computational complexity of any iterative numerical method purely depends upon the total number of arithmetical operations per iteration. The higher the computational complexity of iterative method, the more dense the algorithm indicating the slowness in convergence. In order to measure the computational complexity of both the methods, assume that the solution domain is discretized with the grid size n , then the internal grid points will be λ^2 , where $\lambda = n - 1$. We divide the internal grid points λ^2 of the solution domain into two types of points, namely iterative points and direct points. The iterative points are the points that take a part in getting the convergence of iterative process, while the direct points are the points which can be directly evaluated from the standard grid Crank–Nicolson finite difference formula. Tables 1 and 2 show the computational complexity analysis of both FSG(C-N) and FSkG(C-N) iterative methods before and after convergence respectively. In the FSG(C-N) iterative method, all the internal grid points are the iterative points that utilize all the grid points of the solution domain to achieve the convergence, while in the FSkG(C-N) iterative method, half of the internal grid points are the iterative points that take part for convergence. Once convergence is achieved, the values at the remaining half grid points (direct points) can be calculated by standard difference formula. Table 3 shows the total number of arithmetical operations required per iteration after convergence for each method. For more about the computational complexity of the fractional skewed iterative scheme, please refer to [53].

The maximum error of the exact analytic solution and numerical approximation solution is defined as follows:

$$Max\ Error = \max_{0 \leq i, j \leq n} \max_{0 \leq k \leq l} \{|U_{ij}^k - u_{ij}^k|\}.$$

Example 1 Here we apply both standard and skewed grid numerical schemes on the two-dimensional hyperbolic telegraph equation of fractional order [31]:

$$\frac{\partial^{2\alpha} u}{\partial t^{2\alpha}} + \frac{\partial^\alpha u}{\partial t^\alpha} + u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + f(x, y, t).$$

The initial and boundary conditions are given by

$$\begin{aligned} u(x, y, 0) &= \phi(x, y) = \sin(x) \sin(y), & u_t(x, y, 0) &= 0, \\ u(0, y, t) &= g_1(y, t) = 0, & u(1, y, t) &= g_2(1, y, t) = \cos(t) \sin(1) \sin(y), \\ u(x, 0, t) &= g_3(x, t) = 0, & u(x, 1, t) &= g_4(x, 1, t) = \cos(t) \sin(x) \sin(1), \end{aligned}$$

$$0 \leq x, y \leq 1, 0 \leq t \leq T.$$

Here, $f(x, y, t) = 2 \sin(x) \sin(y)(\cos(t) - \sin(t))$ and the exact analytical solution is $u(x, y, t) = \cos(t) \sin(x) \sin(y)$.

In the solution of the problem, we discretize the solution domain for various mesh sizes of 10, 15, 20, and 25 by assuming $h = \Delta x = \Delta y$ in both x and y directions for the space discretization and for time discretization ($0 < T < 1$), we assume $\tau = 1/10$. Table 4 describes the summary of calculated values of elapsed time (in seconds), number of iterations (Ite), maximum absolute error (Max Error), average absolute error (Ave Error), and the total number of arithmetic operations (Total Operations) by both FSG(C-N) and FSkG(C-N) iterative schemes when $\alpha = 0.65, 0.75, 0.85$. The numerical results show that the FSkG(C-N) iterative method has (30.0 – 38.80)% less computational complexity than the FSG(C-N) iterative method, whereas the computing time and the number of iterations were calculated almost (31.20 – 42.50)% and (58.60 – 65.60)% less respectively, with the same degree of accuracy. The comparison between two iterative schemes is sketched in Figs. 4, 5, 6 when $\alpha = 0.65$, while the graphical representation of the numerical solution is illustrated in Figs. 7, 8, 9 when $\alpha = 0.65, 0.75, 0.85$ respectively.

Example 2 Here we apply both standard and skewed grid numerical schemes on the two-dimensional hyperbolic telegraphic equation of fractional order [31]:

$$\frac{\partial^{2\alpha} u}{\partial t^{2\alpha}} + \frac{\partial^\alpha u}{\partial t^\alpha} + u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + f(x, y, t).$$

The initial and boundary conditions are given by

$$\begin{aligned} u(x, y, 0) &= \phi(x, y) = \sinh(x) \sinh(y), & u_t(x, y, 0) &= 0, \\ u(0, y, t) &= g_1(y, t) = 0, & u(1, y, t) &= g_2(1, y, t) = \cos(t) \sinh(1) \sinh(y), \\ u(x, 0, t) &= g_3(x, t) = 0, & u(x, 1, t) &= g_4(x, 1, t) = \cos(t) \sinh(x) \sinh(1). \end{aligned}$$

Table 4 Comparison between FSG(C-N) and FSkG(C-N) iterative methods at $\tau = 1/10$ for Example 1

h^{-1}	Method	Execution time (sec.)	lte.	Ave error	Max error	Total operations
$\alpha = 0.65$						
10	FSG(C-N)	2.93282	20	4.71168×10^{-2}	9.69517×10^{-2}	403,380
	FSkG(C-N)	1.10761	12	4.88586×10^{-2}	1.00569×10^{-1}	132,468
15	FSG(C-N)	12.5737	35	4.41042×10^{-2}	9.71679×10^{-2}	1,708,140
	FSkG(C-N)	4.30563	21	4.48098×10^{-2}	9.87645×10^{-2}	539,334
20	FSG(C-N)	35.2718	54	4.26392×10^{-2}	9.72095×10^{-2}	4,854,006
	FSkG(C-N)	11.6689	32	4.30456×10^{-2}	9.81329×10^{-2}	1,487,028
25	FSG(C-N)	78.9365	75	4.17918×10^{-2}	9.75259×10^{-2}	10,756,800
	FSkG(C-N)	25.1786	44	4.20595×10^{-2}	9.81334×10^{-2}	3,232,394
$\alpha = 0.75$						
10	FSG(C-N)	2.26201	16	3.82775×10^{-2}	7.67695×10^{-2}	322,704
	FSkG(C-N)	0.90480	10	3.98710×10^{-2}	8.01500×10^{-2}	112,050
15	FSG(C-N)	9.42246	27	3.58042×10^{-2}	7.83929×10^{-2}	1,317,708
	FSkG(C-N)	3.33842	17	3.64650×10^{-2}	7.98128×10^{-2}	441,228
20	FSG(C-N)	26.3018	41	3.46193×10^{-2}	7.86265×10^{-2}	3,685,449
	FSkG(C-N)	8.81406	25	3.49882×10^{-2}	7.94765×10^{-2}	1,171,545
25	FSG(C-N)	57.9856	58	3.39374×10^{-2}	7.85311×10^{-2}	8,318,592
	FSkG(C-N)	18.6265	34	3.41800×10^{-2}	7.91150×10^{-2}	2,514,029
$\alpha = 0.85$						
10	FSG(C-N)	1.87201	13	2.39638×10^{-2}	5.27844×10^{-2}	262,197
	FSkG(C-N)	0.79561	9	2.53145×10^{-2}	5.55785×10^{-2}	101,841
15	FSG(C-N)	7.36325	21	2.24018×10^{-2}	5.21491×10^{-2}	1,024,884
	FSkG(C-N)	2.71442	13	2.29590×10^{-2}	5.34259×10^{-2}	343,122
20	FSG(C-N)	20.1397	32	2.17059×10^{-2}	5.27656×10^{-2}	2,876,448
	FSkG(C-N)	6.89524	19	2.20033×10^{-2}	5.34685×10^{-2}	901,131
25	FSG(C-N)	44.0391	44	2.13271×10^{-2}	5.25357×10^{-2}	6,310,656
	FSkG(C-N)	14.6017	26	2.15153×10^{-2}	5.29944×10^{-2}	1,939,337

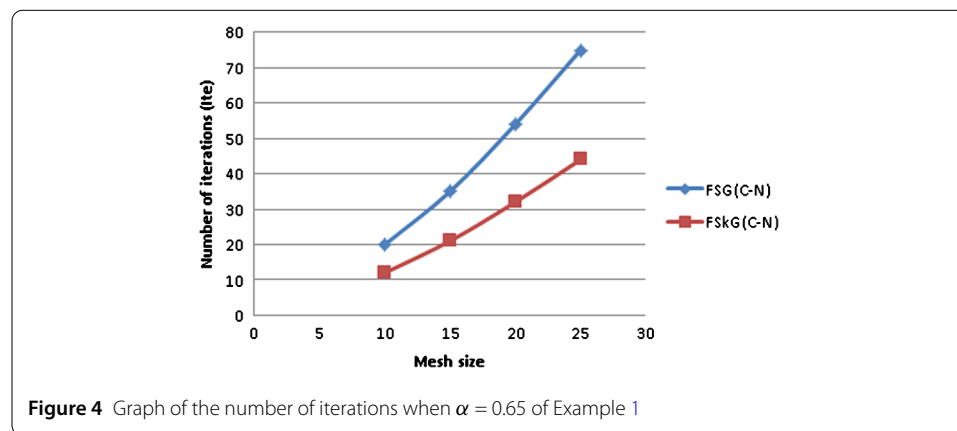


Figure 4 Graph of the number of iterations when $\alpha = 0.65$ of Example 1

Here $f(x, y, t) = \sinh(x) \sinh(y)(\beta^2 \cos(t) - 2\delta \sin(t) - 3 \cos(t))$ and the exact analytical solution is $u(x, y, t) = \cos(t) \sinh(x) \sinh(y)$. Let $\beta = 5$ and $\delta = 10$ in $f(x, y, t)$, in solving the above 2D-HTFDE.

In this example, we discretize the solution domain for various mesh sizes of 12, 16, 20, and 24 by assuming $h = \Delta x = \Delta y$ in both x and y directions for the space discretization and for time discretization ($0 < T < 1$), we assume $\tau = 1/15$. Table 5 describes the summary

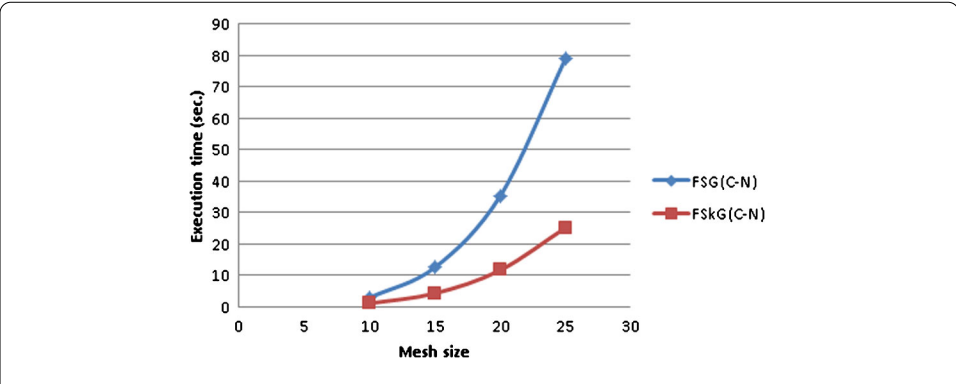


Figure 5 Graph of execution time (sec.) when $\alpha = 0.65$ of Example 1

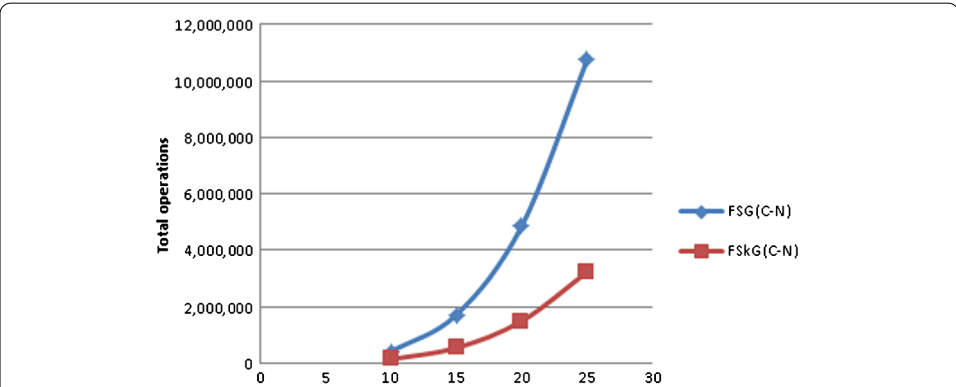


Figure 6 Graph of the total number of operations when $\alpha = 0.65$ of Example 1

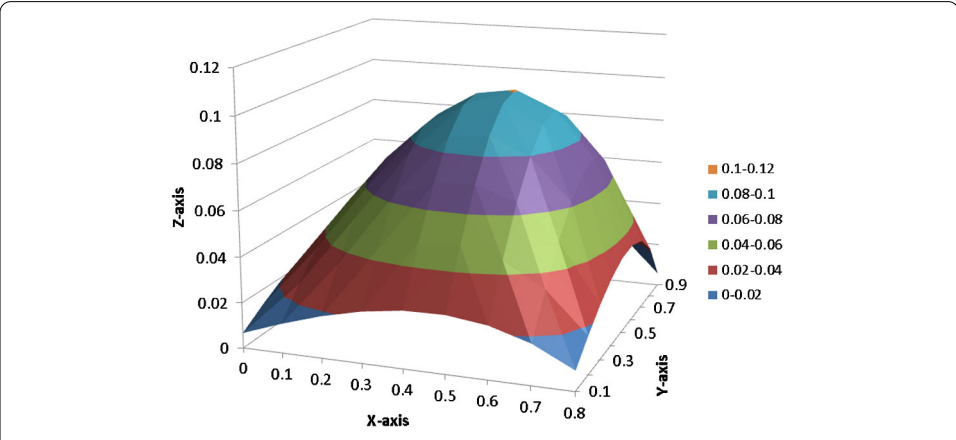
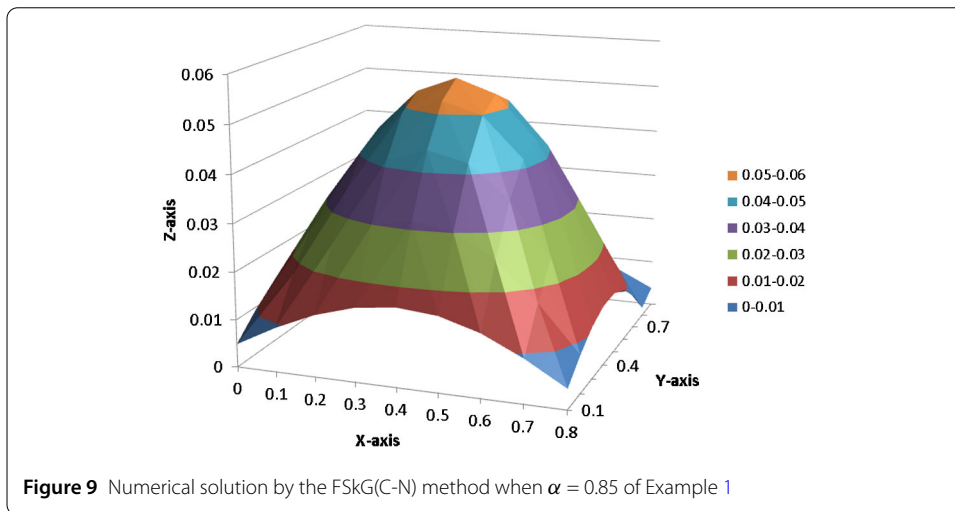
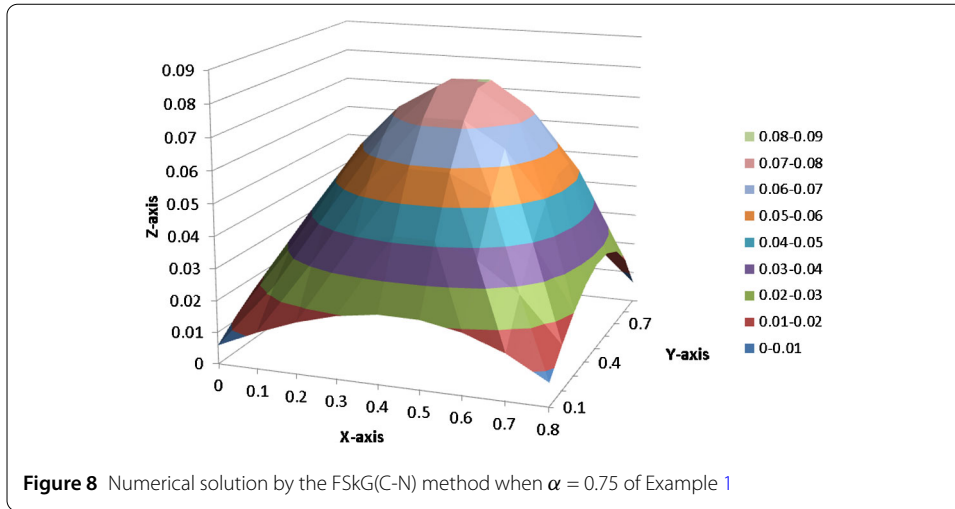


Figure 7 Numerical solution by the FSkG(C-N) method when $\alpha = 0.65$ of Example 1

of calculated values of elapsed time (in seconds), number of iterations (Ite), maximum absolute error (Max Error), average absolute error (Ave Error), and the total number of arithmetic operations (Total Operations) by both FSG(C-N) iterative and FSkG(C-N) iterative schemes when $\alpha = 0.60, 0.70, 0.80$. The numerical results show that the FSkG(C-N) iterative method has (29.70 – 39.50)% less computational complexity than the FSG(C-N)



iterative method, whereas the computing time and the number of iterations were calculated almost (31.80 – 40.80)% and (57.90 – 71.40)% less respectively, with the same degree of accuracy. The comparison between two iterative schemes is sketched in Figs. 10, 11, 12 when $\alpha = 0.70$ and the graphical representation of the numerical solution is illustrated in Figs. 13, 14, 15 when $\alpha = 0.60, 0.70, 0.80$ respectively.

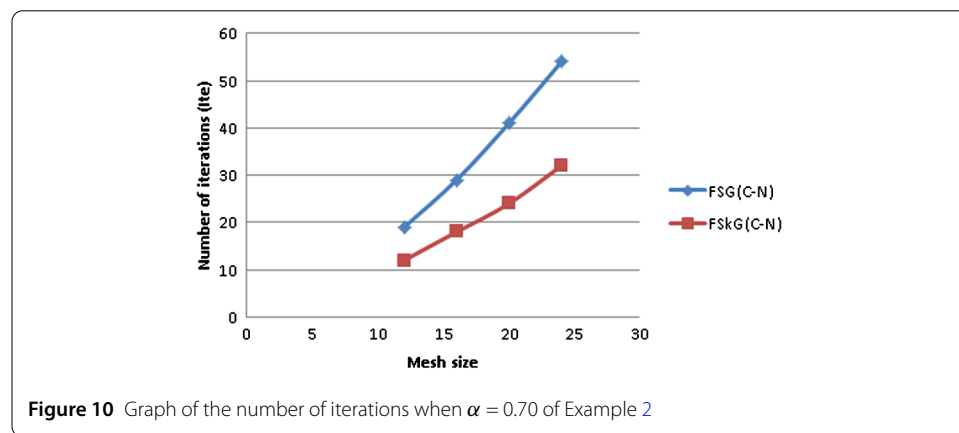
Figures 4–6 and 10–12 show the graphs of the number of iterations, execution of timings (in seconds), and the total number of operations of FSkG(C-N) and FSG(C-N) schemes. In each graph the values of the FSkG(C-N) scheme are significantly less than those of the FSG(C-N) scheme at different values of α in two different examples. Due to this reason, the FSkG(C-N) iterative method has less computational complexity than the FSG(C-N) iterative method, whereas Figs. 7–9 and 13–15 show the graphs of 3D diagram of the numerical solutions of FSkG(C-N) method and variation of graphs can be seen as the values of α varies.

8 Conclusion

In this work, we successfully formulate a new fractional skewed grid point Crank–Nicolson iterative scheme at the point $(x_i, y_j, t_{k+1/2})$ derived from fractional standard five

Table 5 Comparison between FSG(C-N) and FSkG(C-N) iterative methods $\tau = 1/15$ for Example 2

h^{-1}	Method	Execution time (sec.)	lte	Ave error	Max error	Total operations
$\alpha = 0.60$						
12	FSG(C-N)	12.3553	26	1.13977×10^{-2}	3.04174×10^{-2}	1,145,144
	FSkG(C-N)	4.27443	16	1.12180×10^{-2}	3.11192×10^{-2}	377,104
16	FSG(C-N)	34.7570	40	1.12653×10^{-2}	3.04467×10^{-2}	3,276,000
	FSkG(C-N)	11.0605	24	1.11831×10^{-2}	3.08253×10^{-2}	1,027,936
20	FSG(C-N)	78.3905	57	1.11666×10^{-2}	3.08420×10^{-2}	7,490,028
	FSkG(C-N)	25.7246	33	1.11369×10^{-2}	3.09775×10^{-2}	2,239,692
24	FSG(C-N)	150.182	76	1.10778×10^{-2}	3.09794×10^{-2}	14,634,256
	FSkG(C-N)	48.0327	44	1.10844×10^{-2}	3.11293×10^{-2}	4,340,336
$\alpha = 0.70$						
12	FSG(C-N)	8.82966	19	8.23924×10^{-3}	2.59719×10^{-2}	836,836
	FSkG(C-N)	3.26042	12	8.21954×10^{-3}	2.65085×10^{-2}	288,288
16	FSG(C-N)	24.1490	29	8.06517×10^{-3}	2.63727×10^{-2}	2,375,100
	FSkG(C-N)	8.40845	18	8.06394×10^{-3}	2.66746×10^{-2}	781,144
20	FSG(C-N)	53.4459	41	7.96695×10^{-3}	2.65627×10^{-2}	5,387,564
	FSkG(C-N)	17.5189	24	7.98367×10^{-3}	2.67728×10^{-2}	1,646,736
24	FSG(C-N)	104.567	54	7.89936×10^{-3}	2.66550×10^{-2}	10,398,024
	FSkG(C-N)	32.7290	32	7.93508×10^{-3}	2.68291×10^{-2}	3,182,816
$\alpha = 0.80$						
12	FSG(C-N)	6.61444	14	1.15550×10^{-2}	2.71116×10^{-2}	616,616
	FSkG(C-N)	2.69882	10	1.15107×10^{-2}	2.76285×10^{-2}	243,880
16	FSG(C-N)	17.3941	21	1.11922×10^{-2}	2.74914×10^{-2}	1,719,900
	FSkG(C-N)	6.55204	13	1.11648×10^{-2}	2.76485×10^{-2}	575,484
20	FSG(C-N)	37.9550	28	1.09646×10^{-2}	2.75706×10^{-2}	3,679,312
	FSkG(C-N)	13.2445	18	1.09317×10^{-2}	2.76839×10^{-2}	1,251,432
24	FSG(C-N)	72.2909	37	1.08101×10^{-2}	2.76601×10^{-2}	7,124,572
	FSkG(C-N)	23.9930	22	1.07676×10^{-2}	2.76683×10^{-2}	2,218,216



point Crank–Nicolson finite difference approximation by skewing clockwise an angle of 45° with respect to the standard grids in solving the two-dimensional second-order hyperbolic telegraph equation of fractional order. The derived numerical iterative scheme is unconditionally stable, and its stability and convergence are proven by the Fourier analysis. A comparison of both the numerical iterative schemes has been given in terms of computational complexities of the algorithm and verified with the help of examples. Experimental numerical results show that the FSkG(C-N) iterative method not only has the

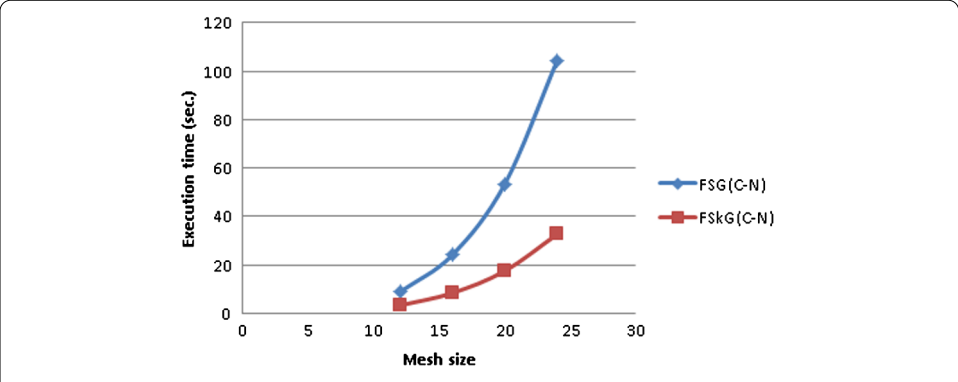


Figure 11 Graph of execution time (sec.) when $\alpha = 0.70$ of Example 2

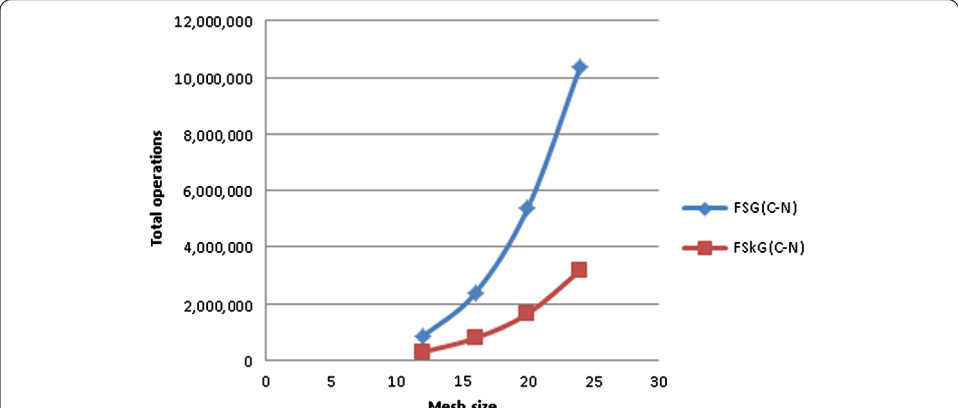


Figure 12 Graph of total operations when $\alpha = 0.70$ of Example 2

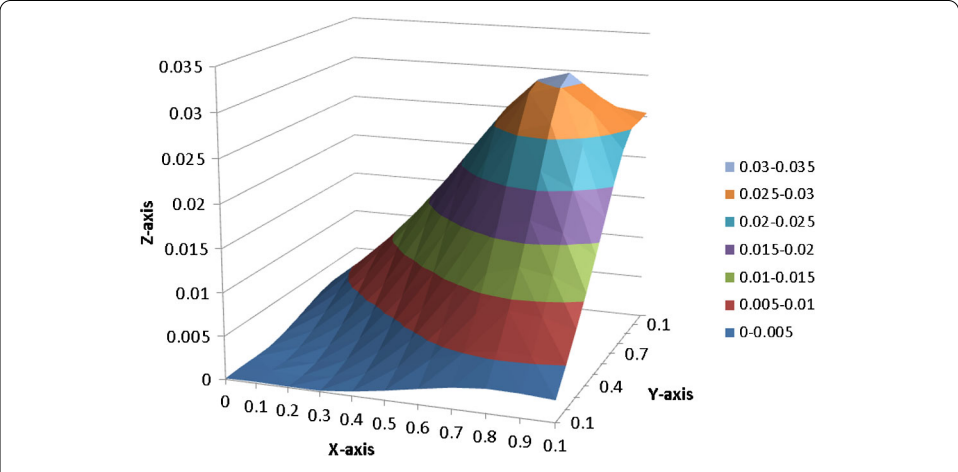
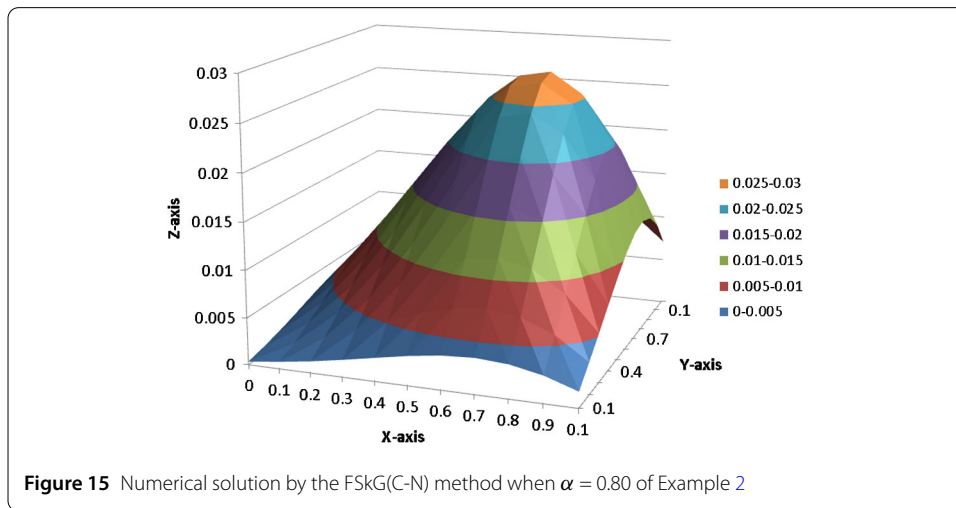
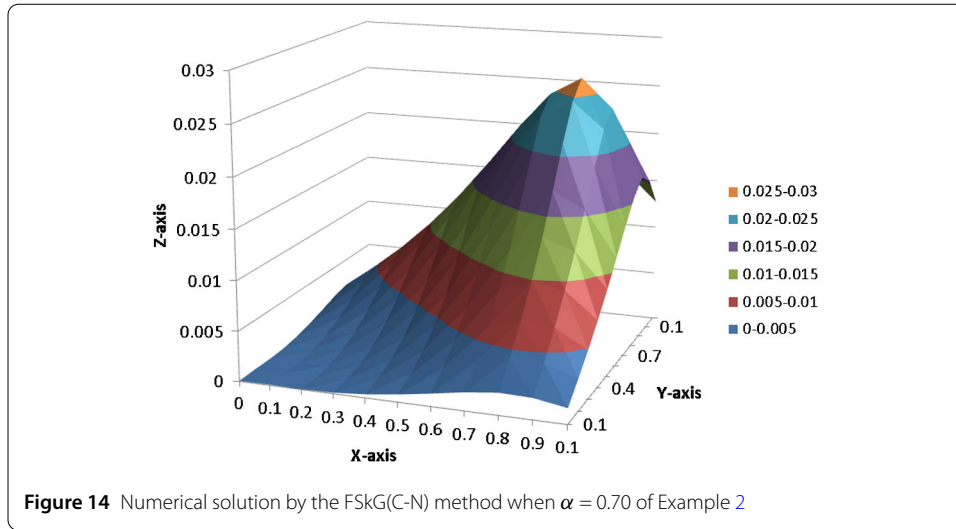


Figure 13 Numerical solution by the FskG(C-N) method when $\alpha = 0.60$ of Example 2

least number of arithmetical operations (per iteration) in terms of computational complexity, but it also requires least CPU execution timings (in seconds) and number of iterations (Ite) when compared to the FSG(C-N) iterative method without jeopardizing the solution accuracies.



Acknowledgements

The authors gratefully acknowledge the financial support from Universiti Sains Malaysia Research University Grants (1001/PMATHS/8011016). The authors are very thankful to the reviewers for the valuable comments and suggestions for improving the quality and standard of the paper.

Funding

All the financial aid for publishing this manuscript is funded by Universiti Sains Malaysia.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

Both authors declare that they have reviewed and approved the final manuscript for publication.

Authors' information

Ajmal Ali is PhD student. Norhashidah Hj. Mohd. Ali is Professor.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 10 January 2019 Accepted: 14 July 2019 Published online: 25 July 2019

References

1. Bagley, R.L., Torvik, P.J.: On the appearance of the fractional derivative in the behavior of real materials. *J. Appl. Mech.* **51**, 294–298 (1984)
2. Mainardi, F.: *Fractals and Fractional Calculus Continuum Mechanics*. Springer, Bologna (1997)
3. Herrmann, R.: *Fractional Calculus: An Introduction for Physicists*. World Scientific, Hackensack (2011)
4. Benson, D.A., Wheatcraft, S.W.: Application of a fractional advection-dispersion equation. *Water Resour. Res.* **36**(6), 1403–1412 (2000)
5. Bagley, R.L., Torvik, P.J.: Theoretical basis for the application of fractional calculus to viscoelasticity. *J. Rheol.* **27**(3), 201–210 (2013)
6. Mainardi, F.: *Fractional Calculus and Waves in Linear Viscoelasticity: An Introduction to Mathematical Models*. Imperial College Press, London (2010)
7. Magin, R.L.: *Fractional Calculus in Bioengineering*. Begell House Publishers, Danbury (2006)
8. Agrawal, O.P., Defterli, O., Baleanu, D.: Fractional optimal control problems with several state and control variables. *J. Vib. Control* **16**(13), 1967–1976 (2010)
9. Moshrefi, M., Hammond, J.K.: Physical and geometrical interpretations of fractional operators. *J. Franklin Inst.* **335**(5), 1077–1086 (1998)
10. Zhou, J., Pu, Y., Liao, K.: *Fractional Order Calculus Principle and Its Application in Latest Single Analysis and Processing*. Beijing Science Press, Beijing (2010)
11. Podlubny, I.: *Fractional Differential Equations*. Academic Press, New York (1999)
12. Singh, J., Kumara, D., Baleanu, D., Rathore, S.: An efficient numerical algorithm for the fractional Drinfeld–Sokolov–Wilson equation. *Appl. Math. Comput.* **335**, 12–24 (2018)
13. Kumar, D., Singh, J., Baleanu, D., Rathore, S.: Analysis of a fractional model of the Ambartsumian equation. *Eur. Phys. J. Plus* **133**, 259 (2018)
14. Jajarmi, A., Baleanu, D.: Suboptimal control of fractional-order dynamic systems with delay argument. *J. Vib. Control* **24**(12), 2430–2446 (2018)
15. Yang, X.J.: *Advanced Local Fractional Calculus and Its Applications*. World Science Publisher, New York (2012)
16. Singh, J., Kumar, D., Baleanu, D., Rathore, S.: On the local fractional wave equation in fractal strings. *Math. Methods Appl. Sci.* **42**(5), 1588–1595 (2019)
17. Yang, X.J., Machado, T.A.J., Baleanu, D.: Exact traveling-wave solution for local fractional Boussinesq equation in fractal domain. *Fractals* **25**(4), 1740006 (2017)
18. Kumar, D., Tchier, F., Singh, J., Baleanu, D.: An efficient computational technique for fractal vehicular traffic flow. *Entropy* **20**, 259 (2018)
19. Sun, H.G., Chen, W., Wei, H., Chen, Y.Q.: A comparative study of constant-order and variable-order fractional models in characterizing memory property of systems. *Eur. Phys. J. Spec. Top.* **193**, 185 (2011)
20. Hajjipour, M., Jajarmi, A., Baleanu, D., Sun, H.G.: On an accurate discretization of a variable-order fractional reaction-diffusion equation. *Commun. Nonlinear Sci. Numer. Simul.* **69**, 119–133 (2019)
21. Meng, R., Yin, D., Drapaca, C.S.: Variable-order fractional description of compression deformation of amorphous glassy polymers. *Comput. Mech.* **64**(1), 163–171 (2019)
22. Cao, J., Qiu, Y., Song, G.: A compact finite difference scheme for variable order subdiffusion equation. *Commun. Nonlinear Sci. Numer. Simul.* **48**, 140–149 (2017)
23. Jajarmi, A., Baleanu, D.: A new fractional analysis on the interaction of HIV with T-cells. *Chaos Solitons Fractals* **113**, 221–229 (2018)
24. Baleanu, D., Jajarmi, A., Bonyah, E., Hajjipour, M.: New aspects of poor nutrition in the life cycle within the fractional calculus. *Adv. Differ. Equ.* **2018**, 230 (2018)
25. Kumar, D., Singh, J., Baleanu, D., Rathore, S.: Analysis of regularized long-wave equation associated with a new fractional operator with Mittag-Leffler type kernel. *Phys. A, Stat. Mech. Appl.* **492**, 155–167 (2018)
26. Baleanu, D., Jajarmi, A., Hajjipour, M.: On the nonlinear dynamical systems within the generalized fractional derivatives with Mittag-Leffler kernel. *Nonlinear Dyn.* **94**, 397–414 (2018)
27. Shamasneh, A.R., Jalab, H.A., Palaiahnakote, S., Obaidallah, U.H., Ibrahim, R.W., Melegy, M.T.: A new local fractional entropy-based model for kidney MRI image enhancement. *Entropy* **20**, 344 (2018)
28. Raftari, B., Yildirim, A.: Analytical solution of second-order hyperbolic telegraph equation by variational iteration and homotopy perturbation methods. *Results Math.* **61**, 13–28 (2010)
29. Dehghan, M., Mehebbi, A.: A high order implicit collocation method for the solution of two-dimensional linear hyperbolic equation. *Numer. Methods Partial Differ. Equ.* **25**, 232–243 (2009)
30. Srivastava, V.K., Awasthi, M.K., Chaurasia, R.K.: Reduced differential transform method to solve two and three dimensional second order hyperbolic telegraph equations. *J. King Saud Univ., Eng. Sci.* **29**, 166–171 (2017)
31. Mittal, R.C., Bhatia, R.: A numerical study of two dimensional hyperbolic telegraph equation by modified B-spline differential quadrature method. *Appl. Math. Comput.* **244**, 976–997 (2014)
32. Mohanty, R.K., Jain, M.K.: An unconditionally stable alternating direction implicit scheme for the two space dimensional linear hyperbolic equation. *Numer. Methods Partial Differ. Equ.* **7**, 684–688 (2001)
33. Mohanty, R.K., Jain, M.K., Arora, U.: An unconditionally stable ADI method for the linear hyperbolic equation in three space dimensional. *Int. J. Comput. Math.* **79**, 133–142 (2002)
34. Mohanty, R.K.: A new unconditionally stable difference schemes for the solution of multi-dimensional telegraphic equations. *Int. J. Comput. Math.* **86**(12), 2061–2071 (2009)
35. Karaa, S.: Unconditionally stable ADI scheme of higher-order for linear hyperbolic equations. *Int. J. Comput. Math.* **87**(13), 3030–3038 (2010)
36. Ding, H., Zhang, Y.: A new fourth-order compact finite difference scheme for the two-dimensional second-order hyperbolic equation. *J. Comput. Appl. Math.* **230**(2), 626–632 (2009)
37. Jiwari, R., Pandit, S., Mittal, R.C.: A differential quadrature algorithm to solve the two dimensional linear hyperbolic telegraph equation with Dirichlet and Neumann boundary conditions. *Appl. Math. Comput.* **218**, 7279–7294 (2012)
38. Wu, J.: *Theory and Applications of Partial Functional Differential Equations*. Springer, New York (1996)
39. Shivanian, E.: Spectral meshless radial point interpolation method to two-dimensional fractional telegraph equation. *Math. Methods Appl. Sci.* **39**(7), 1820–1835 (2016)

40. Ferreira, M., Rodrigues, M., Vieira, N.: Fundamental solution of the multi-dimensional time fractional telegraph equation. *Fract. Calc. Appl. Anal.* **20**(4), 868–894 (2017)
41. Shivanian, E., Abbasbandy, S., Alhuthali, M.S., Alsulami, H.H.: Local integration of 2-D fractional telegraph equation via moving least squares approximation. *Eng. Anal. Bound. Elem.* **56**, 98–105 (2015)
42. Heydari, M.H., Hooshmandasl, R.M., Mohammadi, F.: Two-dimensional Legendre wavelets for solving time-fractional telegraph equation. *Adv. Appl. Math. Mech.* **6**(2), 247–260 (2014)
43. Hosseini, V.R., Shivanian, E., Chen, W.: Local integration of 2-D fractional telegraph equation via local radial point interpolant approximation. *Eur. Phys. J. Plus* **130**(2), 33 (2015)
44. Abdullah, A.R.: The four point explicit de-coupled group (EDG) method: a fast Poisson solver. *Int. J. Comput. Math.* **38**, 61–70 (1991)
45. Ali, N.H.M., Fu, N.K.: Modified explicit de-coupled group method in the solution of 2D elliptic PDEs. In: Abstracts of the 12th WSEAS International Conference on Applied Mathematics, 29–31 December, Cairo, Egypt (2007)
46. Evans, D.: Group explicit iterative methods for solving large linear systems. *Int. J. Comput. Math.* **17**(1), 81–108 (1985)
47. Evans, D., Yousif, W.: Explicit group iterative methods for solving elliptic partial differential equations in 3-space dimensions. *Int. J. Comput. Math.* **18**, 323–340 (1986)
48. Kew, L.M., Ali, N.H.M.: New explicit group iterative methods in the solution of three dimensional hyperbolic telegraph equations. *J. Comput. Phys.* **294**, 382–404 (2015)
49. Kew, L.M., Ali, N.H.M.: Explicit group iterative methods for the solution of telegraph equations. In: Abstracts of the World Congress on Engineering, 30 Jun–2 Jul, London, United Kingdom (2010)
50. Ali, N.H.M., Kew, L.M.: New explicit group iterative methods in the solution of two dimensional hyperbolic equations. *J. Comput. Phys.* **231**, 6953–6968 (2012)
51. Othman, M., Abdullah, A.: An efficient four points modified explicit group Poisson solver. *Int. J. Comput. Math.* **76**(2), 203–217 (2000)
52. Yousif, W., Evans, D.J.: Explicit de-coupled group iterative methods and their parallel implementations. *Parallel Algorithms Appl.* **7**, 53–71 (1995)
53. Balasim, A.T., Ali, N.H.M.: A rotated Crank–Nicolson iterative method for the solution of two-dimensional time-fractional diffusion equation. *Indian J. Sci. Technol.* **8**(32), 1–8 (2015)
54. Balasim, A.T., Ali, N.H.M.: Group iterative methods for the solution of two-dimensional time-fractional diffusion equation. In: Abstracts of the 23rd Malaysian National Symposium of Mathematical Sciences, 24–26 November, Johor Bahru, Malaysia (2015)
55. Balasim, A.T., Ali, N.H.M.: The solution of 2-D time-fractional diffusion equation by fractional modified explicit group iterative method. In: Abstracts of the 2nd International Conference on Mathematics, Engineering and Industrial Applications, 10–12 August, Songkhla, Thailand (2016)
56. Ali, U., Abdullah, F.A., Din, S.T.M.: Modified implicit fractional difference scheme for 2D modified anomalous fractional sub-diffusion equation. *Adv. Differ. Equ.* **2017**, 185 (2017)

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)
