


RESEARCH

Open Access



# Efficient hybrid group iterative methods in the solution of two-dimensional time fractional cable equation

Fouad Mohammad Salama<sup>1\*</sup> , Norhashidah Hj. Mohd. Ali<sup>1</sup> and Nur Nadiah Abd Hamid<sup>1</sup>

\*Correspondence:

[fuadmohd321@gmail.com](mailto:fuadmohd321@gmail.com)

<sup>1</sup>School of Mathematical Sciences,  
Universiti Sains Malaysia, Penang,  
Malaysia

## Abstract

In this paper, the development of new hybrid group iterative methods for the numerical solution of a two-dimensional time-fractional cable equation is presented. We use Laplace transform method to approximate the time fractional derivative which reduces the problem into an approximating partial differential equation. The obtained partial differential equation is solved by four-point group iterative methods derived from two implicit finite difference schemes. Matrix norm analysis together with mathematical induction are utilized to investigate the stability and convergence properties. A comparative study with the recently developed hybrid standard point (HSP) iterative method accompanied by their computational cost analysis are also given. Numerical experiments are conducted to demonstrate the superiority of the proposed hybrid group iterative methods over the HSP iterative method in terms of the number of iterations, computational cost as well as the CPU times.

**Keywords:** Caputo fractional derivative; Time-fractional derivative; Cable equation; Laplace transform; Finite differences; Group iterative methods; Stability and convergence analyses

## 1 Introduction

The numerical solutions of fractional differential equations are of great importance in describing and modeling many problems in engineering and applied sciences. In this study, we consider the following two-dimensional problem of a class of time-fractional cable type,

$${}_0^C D_t^\alpha u(x, y, t) = a_x \frac{\partial^2 u(x, y, t)}{\partial x^2} + a_y \frac{\partial^2 u(x, y, t)}{\partial y^2} - \mu_0 u(x, y, t) + f(x, y, t), \quad 0 < \alpha < 1, \quad (1)$$

subject to the initial and boundary conditions

$$u(x, y, 0) = p(x, y), \quad (2)$$

$$u(x, 0, t) = p_1(x, t), \quad u(x, L, t) = p_2(x, t), \quad (3)$$

$$u(0, y, t) = p_3(y, t), \quad u(L, y, t) = p_4(y, t),$$

© The Author(s) 2020. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

defined on  $\Omega = \{(x, y, t) | 0 \leq x, y \leq L, 0 \leq t \leq T\}$ , where  $a_x, a_y$  and  $\mu_0$  are positive constants. Here,  $0 < \alpha < 1$  is the order of the Caputo fractional derivative defined by

$${}_0^C D_t^\alpha u(x, y, t) = \frac{1}{\Gamma(n - \alpha)} \int_0^t \frac{u^{(n)}(x, y, \tau)}{(t - \tau)^{\alpha+1-n}} d\tau.$$

Fractional cable equations play a crucial role in modeling anomalous diffusion in spiny neuronal dendrites in biological systems [1]. Since the equations containing fractional order derivatives are deeply complex, and not easy to solve analytically, it is recommended to investigate their solutions numerically. In recent years, a variety of numerical methods including finite element, finite difference and collocation methods have been established for solving one-dimensional and two-dimensional fractional cable equations [1–9]. For instance, an unconditionally stable compact finite difference scheme with convergence order  $O(\tau + h^4)$  has been suggested to generate highly accurate results [2] for one-dimensional fractional cable equation. Zhang et al. [3] established the numerical method in solving the two-dimensional fractional cable equation using collocation and finite difference methods for the space and time discretizations, respectively. Liu et al. [4] developed a numerical scheme based on finite element in space and finite difference in time for solving one-dimensional and two-dimensional time-fractional cable equations. They proved that the resulting scheme is unconditionally stable and the convergence order is  $O(\tau^{\min\{1+\alpha_1, 1+\alpha_2\}} + h^{r+1})$ . Yo and Jiang [5] presented a compact finite difference scheme of fourth order accuracy for solving two-dimensional fractional cable equation. Later, Li et al. [6] formulated another compact difference scheme with better accuracy in time for the two-dimensional fractional cable equation. They proved that the compact scheme is unconditionally stable and the numerical solution converges to the exact solution with order  $O(\tau^2 + h_x^4 + h_y^4)$ . In [9], Li and Rui presented an unconditionally stable block-centered finite difference method for solving the non-linear fractional cable equation on non-uniform grid. In another study, Sweilam and Al-Mekhlafi [8] proposed a new fractional cable equation in which the fractional operator is described in the Atangana–Baleanu–Caputo sense. The Atangana–Baleanu derivative has been employed in describing many fractional problems very recently [10–12]. A non-standard compact finite difference scheme is formulated to solve the resulting problem.

In solving differential equations numerically, the complexity of fractional differential equations is well-known to be significantly greater than that of integer order differential equations. The discretization of differential operators with integer and non-integer orders is the fundamental base of almost all numerical schemes proposed in the literature so far, see [13] and the literature therein. Numerical methods based on discretization schemes for solving time-fractional partial differential equations require total  $O(MN^2)$  computational cost and  $O(MN)$  memory complexity, compared with  $O(MN)$  cost and  $O(M)$  memory for integer order partial differential equations [14–16], where  $N$  and  $M$  are the total number of time levels and spatial grid points, respectively. This is mainly caused by the non-local property of the fractional operator that necessitates the storage of all the preceding solutions to compute the solution at the present time level, making the computations even more complicated and very expensive in terms of the memory and CPU time usage. In the light of such computational challenges in solving time-fractional differential equations, developing efficient numerical methods that generate fast results and use less computer resources is of great importance. Therein lies the main motivation of this study. In regard

to two-dimensional time-fractional cable equation, fast and unconditionally stable numerical schemes are quite rare in the literature. An example of such scarcities is a method presented by Liu et al. [7] who formulated the high order compact difference scheme for solving time-fractional cable equation. The Riemann–Liouville fractional derivative was used to approximate the time derivative. In the same study, the authors have employed the fast Fourier transform method to accelerate their compact scheme, where the computational cost has been reduced to  $O(MN \log^2 N)$ . Recently, Salama and Ali [17] developed fast hybrid standard point (HSP) iterative method based on a combination of Laplace transform method and implicit finite difference scheme for solving the two-dimensional time-fractional cable equation (1). It has been proven that the HSP method is unconditionally stable, and it performs much faster than an existing standard finite difference scheme as it requires only  $O(MN)$  computational cost and  $O(M)$  memory complexity.

In solving multi-dimensional fractional differential equations, it is worth pointing out that the finite difference discretizations of these equations would result in large and sparse systems of linear equations. Due to the sparsity of the coefficient matrix of the resulting linear systems, iterative methods are viewed as more efficient solvers for such linear systems in comparison with direct methods [18]. Among iterative methods, group iterative schemes derived from standard point finite difference approximations have been widely incorporated in solving the linear systems that emerge from the discretization of various types of partial differential equations, see [18–26] and the literature therein. This interest in grouping strategies mainly attributed to their ability to reduce each of the spectral radius of the iteration matrix and the computing effort required at each iteration [27], making them computationally superior to their corresponding standard point iterative schemes. Due to their promising results in solving integer order partial differential equations, interest is now turned to the formulation of group strategies for solving fractional differential equations. Some attempts have been done recently to solve the two-dimensional time-fractional advection–diffusion equation [28], two-dimensional time-fractional diffusion-wave equation [29] and fractional two-point boundary value problem [30]. However, the development of unconditionally stable group iterative schemes for solving fractional differential equations is still at its infancy. Motivated by this background, the primary contribution of our paper is to develop new hybrid group iterative methods for the numerical solution of the two-dimensional time-fractional cable equation (1). We prove the unconditional stability and convergence of the proposed method via matrix norm analysis. The resulting hybrid group iterative methods generate accurate numerical solutions and reduce the computational cost, iterations number and CPU time significantly compared to the HSP iterative method presented in [17]. To the best of our knowledge, this work has not been done by other researchers.

The rest of this article is structured as follows. In Sect. 2, we provide a brief description of the HSP iterative method for solving problem (1). In Sect. 3, we explain the formulation of the proposed hybrid group iterative methods pursued by stability and convergence analyses in Sect. 4. In order to verify the efficiency of the proposed methods, several computational experiments are conducted and presented with their results in Sect. 5. Finally, we conclude our remarks in Sect. 6.

## 2 Review of the hybrid standard point (HSP) iterative method

The Laplace transform is regarded as a very important transform which can be utilized to solve many models arising in various fields of science, technology and engineering [31].

Due to the presence of the fractional derivative's non-local property, the design of finite difference methods for solving problem (1) necessitates the storage of the solution outcomes at all previous time levels if the solution at the present time level is to be computed. To surmount this obstacle, the Laplace transform method together with the linearization property suggested by Ren et al. [32] were used to approximate the Caputo fractional derivative as follows [17]:

$${}_0^C D_t^\alpha u(x, y, t) \approx \alpha \frac{\partial u(x, y, t)}{\partial t} + (1 - \alpha)[u(x, y, t) - u(x, y, 0)]. \quad (4)$$

By substituting (4) into (1), the original two-dimensional time-fractional cable equation (1) is approximated by the following partial differential equation:

$$\frac{\partial u}{\partial t} = A_x \frac{\partial^2 u(x, y, t)}{\partial x^2} + A_y \frac{\partial^2 u(x, y, t)}{\partial y^2} - \eta u(x, y, t) + (r - 1)p(x, y) + rf(x, y, t), \quad (5)$$

$$u(x, y, 0) = p(x, y), \quad (x, y) \in \Omega, \quad (6)$$

$$u(x, 0, t) = p_1(x, t), \quad u(x, L, t) = p_2(x, t), \quad (7)$$

$$u(0, y, t) = p_3(y, t), \quad u(L, y, t) = p_4(y, t), \quad (x, y, t) \in \Omega,$$

where  $A_x = \frac{a_x}{\alpha}$ ,  $A_y = \frac{a_y}{\alpha}$ ,  $\eta = \frac{1-\alpha+\mu_0}{\alpha}$  and  $r = \frac{1}{\alpha}$  are positive constants.

In numerically solving the original problem (1), an economical computational solution can be obtained by solving the resulting approximating partial differential equation (5) using finite difference methods. For the discretization of the solution domain, we utilize a uniform grid points  $(x_i, y_j, t_k)$ , with  $x_i = ih$ ,  $y_j = jh$ ,  $i, j = 0, 1, \dots, n$ , and  $t_k = k\Delta t$ ,  $k = 0, 1, \dots, N$  for some positive integers  $n$  and  $N$ ,  $h = \Delta x = \Delta y = \frac{L}{n}$  and  $\Delta t = \frac{T}{N}$  are the uniform space and time step sizes, respectively. Various finite difference schemes can be utilized to solve (5). Here, and depending on the discretizations forward in time and centered in space about the point  $(x_i, y_j, t_k)$ , the following HSP iterative scheme is obtained [17]:

$$u_{i,j}^{k+1} = \frac{1}{1 + \eta\Delta t + 2d_1 + 2d_2} [d_1(u_{i+1,j}^{k+1} + u_{i-1,j}^{k+1}) + d_2(u_{i,j+1}^{k+1} + u_{i,j-1}^{k+1}) + u_{i,j}^k + (r-1)\Delta t u_{i,j}^0 + r\Delta t f_{i,j}^{k+1}], \quad (8)$$

where  $d_1 = \frac{A_x \Delta t}{h^2}$ , and  $d_2 = \frac{A_y \Delta t}{h^2}$ . In applying this HSP iterative method, the iteration process at any time level is carried out on all of the solution grid points using Eq. (8) until a predefined convergence criterion is attained, prior to proceeding to the next time level. The process goes on until it hits the target time level.

The advantage of the described HSP method lies in its ability to generate fast numerical solutions by reducing the computational cost and memory requirement significantly in comparison with the standard finite difference schemes used to solve problem (1). For further details, refer to [17]. As the group iterative methods can accelerate the rate of convergence compared to their counterparts point iterative methods, the formulation of the hybrid explicit group (HEG) and the hybrid modified explicit group (HMEG) iterative methods will be illustrated in the next section.

### 3 Design of the hybrid group iterative methods

#### 3.1 The hybrid explicit group (HEG) iterative method

In order to formulate the HEG method, we assume that the grid points of the solution domain at any time level are arranged in group of four points as illustrated in Fig. 1. Then, we apply Eq. (8) to each of these points so that the following  $(4 \times 4)$  system of equations is obtained:

$$\begin{pmatrix} V & -d_1 & 0 & -d_2 \\ -d_1 & V & -d_2 & 0 \\ 0 & -d_2 & V & -d_1 \\ -d_2 & 0 & -d_1 & V \end{pmatrix} \begin{pmatrix} u_{i,j}^{k+1} \\ u_{i+1,j}^{k+1} \\ u_{i+1,j+1}^{k+1} \\ u_{i,j+1}^{k+1} \end{pmatrix} = \begin{pmatrix} rhs_{i,j} \\ rhs_{i+1,j} \\ rhs_{i+1,j+1} \\ rhs_{i,j+1} \end{pmatrix}, \quad (9)$$

where

$$\begin{aligned} V &= 1 + \eta \Delta t + 2d_1 + 2d_2, \\ rhs_{i,j} &= d_1 u_{i-1,j}^{k+1} + d_2 u_{i,j-1}^{k+1} + u_{i,j}^k + (r-1) \Delta t u_{i,j}^0 + r \Delta t f_{i,j}^{k+1}, \\ rhs_{i+1,j} &= d_1 u_{i+2,j}^{k+1} + d_2 u_{i+1,j-1}^{k+1} + u_{i+1,j}^k + (r-1) \Delta t u_{i+1,j}^0 + r \Delta t f_{i+1,j}^{k+1}, \\ rhs_{i+1,j+1} &= d_1 u_{i+2,j+1}^{k+1} + d_2 u_{i+1,j+2}^{k+1} + u_{i+1,j+1}^k + (r-1) \Delta t u_{i+1,j+1}^0 \\ &\quad + r \Delta t f_{i+1,j+1}^{k+1}, \\ rhs_{i,j+1} &= d_1 u_{i-1,j+1}^{k+1} + d_2 u_{i,j+2}^{k+1} + u_{i,j+1}^k + (r-1) \Delta t u_{i,j+1}^0 + r \Delta t f_{i,j+1}^{k+1}. \end{aligned}$$

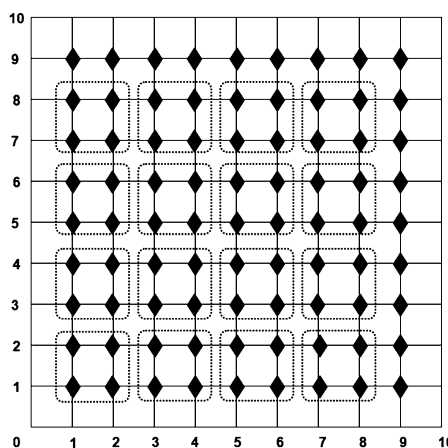
Invert the coefficients matrix in (9) results in the following four-point HEG formula:

$$\begin{pmatrix} u_{i,j}^{k+1} \\ u_{i+1,j}^{k+1} \\ u_{i+1,j+1}^{k+1} \\ u_{i,j+1}^{k+1} \end{pmatrix} = \frac{1}{a} \begin{pmatrix} a_1 & a_2 & a_3 & a_4 \\ a_2 & a_1 & a_4 & a_3 \\ a_3 & a_4 & a_1 & a_2 \\ a_4 & a_3 & a_2 & a_1 \end{pmatrix} \begin{pmatrix} rhs_{i,j} \\ rhs_{i+1,j} \\ rhs_{i+1,j+1} \\ rhs_{i,j+1} \end{pmatrix}, \quad (10)$$

where

$$\begin{aligned} a &= (1 + d_1 + d_2 + \eta \Delta t)(1 + 3d_1 + d_2 + \eta \Delta t)(1 + d_1 + 3d_2 + \eta \Delta t) \\ &\quad \times (1 + 3d_1 + 3d_2 + \eta \Delta t), \\ a_1 &= (1 + 2d_1 + 2d_2 + \eta \Delta t)(1 + 4d_1 + 3d_1^2 + 4d_2 + 8d_1d_2 + 3d_2^2 + 4d_1\eta \Delta t \\ &\quad + 4d_2\eta \Delta t + 2\eta \Delta t + (\eta \Delta t)^2), \\ a_2 &= d_1(1 + 4d_1 + 3d_1^2 + 4d_2 + 8d_1d_2 + 5d_2^2 + 4d_1\eta \Delta t + 4d_2\eta \Delta t + 2\eta \Delta t \\ &\quad + (\eta \Delta t)^2), \\ a_3 &= 2d_1d_2(1 + 2d_1 + 2d_2 + \eta \Delta t), \\ a_4 &= d_2(1 + 4d_1 + 5d_1^2 + 4d_2 + 8d_1d_2 + 3d_2^2 + 4d_1\eta \Delta t + 4d_2\eta \Delta t + 2\eta \Delta t \\ &\quad + (\eta \Delta t)^2). \end{aligned}$$

**Figure 1** Four-point groups of the HEG method with mesh size  $n = 10$



In applying this HEG method, iterations at time level  $t_{k+1}$  are generated on each group of grid points using Eq. (10) until a predefined convergence criterion is attained. The converged solution values are then adopted as the initial guess for the next time level. Through the iteration process, we treat each group of four points explicitly similar to the way we treat the single point in the point iterative methods. The process goes on until it hits the target time level. From Fig. 1, it is worth noting that ungrouped points would take a place near to the top and right boundaries if  $n$  is even. In such a case, the HSP formula (8) is used to iterate the solutions on the ungrouped points next to the boundaries.

### 3.2 The hybrid modified explicit group (HMEG) iterative method

The HMEG method is constructed based on a new uniform grid of step size  $2h = 2L/n$ . By utilizing the forward in time and centered in space discretizations about these  $2h$ -spaced points, the following  $2h$  spacing-based HSP formula is obtained to discretize Eq. (5):

$$\begin{aligned} \frac{u_{ij}^{k+1} - u_{ij}^k}{\Delta t} = & A_x \left( \frac{u_{i+2,j}^{k+1} - 2u_{ij}^{k+1} + u_{i-2,j}^{k+1}}{4h^2} \right) + A_y \left( \frac{u_{i,j+2}^{k+1} - 2u_{ij}^{k+1} + u_{i,j-2}^{k+1}}{4h^2} \right) \\ & - \eta u_{ij}^{k+1} + (r-1)u_{ij}^0 + r f_{ij}^{k+1} + O(\Delta t + (\Delta x)^2 + (\Delta y)^2). \end{aligned} \quad (11)$$

Upon simplification, the above equation can be rewritten as

$$\begin{aligned} u_{ij}^{k+1} = & \frac{1}{1 + \eta \Delta t + d_1/2 + d_2/2} \left[ \frac{d_1}{4} (u_{i+2,j}^{k+1} + u_{i-2,j}^{k+1}) + \frac{d_2}{4} (u_{i,j+2}^{k+1} + u_{i,j-2}^{k+1}) \right. \\ & \left. + u_{ij}^k + (r-1)\Delta t u_{ij}^0 + r \Delta t f_{ij}^{k+1} \right]. \end{aligned} \quad (12)$$

Consider the group of four points  $(i, j)$ ,  $(i+2, j)$ ,  $(i+2, j+2)$  and  $(i, j+2)$  at any time level. Applying Eq. (12) at these four interior points will generate the following  $4 \times 4$  system of equations:

$$\begin{pmatrix} V^* & -d_1/4 & 0 & -d_2/4 \\ -d_1/4 & V^* & -d_2/4 & 0 \\ 0 & -d_2/4 & V^* & -d_1/4 \\ -d_2/4 & 0 & -d_1/4 & V^* \end{pmatrix} \begin{pmatrix} u_{ij}^{k+1} \\ u_{i+2,j}^{k+1} \\ u_{i+2,j+2}^{k+1} \\ u_{i,j+2}^{k+1} \end{pmatrix} = \begin{pmatrix} rhs_{ij} \\ rhs_{i+2,j} \\ rhs_{i+2,j+2} \\ rhs_{i,j+2} \end{pmatrix}, \quad (13)$$

where

$$\begin{aligned}
 V^* &= 1 + \eta \Delta t + (d_1/2) + (d_2/2), \\
 rhs_{ij} &= (d_1/4)u_{i-2,j}^{k+1} + (d_2/4)u_{i,j-2}^{k+1} + u_{ij}^k + (r-1)\Delta t u_{ij}^0 + r\Delta t f_{ij}^{k+1}, \\
 rhs_{i+2,j} &= (d_1/4)u_{i+4,j}^{k+1} + (d_2/4)u_{i+2,j-2}^{k+1} + u_{i+2,j}^k + (r-1)\Delta t u_{i+2,j}^0 \\
 &\quad + r\Delta t f_{i+2,j}^{k+1}, \\
 rhs_{i+2,j+2} &= (d_1/4)u_{i+4,j+2}^{k+1} + (d_2/4)u_{i+2,j+4}^{k+1} + u_{i+2,j+2}^k + (r-1)\Delta t u_{i+2,j+2}^0 \\
 &\quad + r\Delta t f_{i+2,j+2}^{k+1}, \\
 rhs_{ij+2} &= (d_1/4)u_{i-2,j+2}^{k+1} + (d_2/4)u_{i,j+4}^{k+1} + u_{ij+2}^k + (r-1)\Delta t u_{ij+2}^0 \\
 &\quad + r\Delta t f_{ij+2}^{k+1}.
 \end{aligned}$$

By inverting the coefficients matrix in (13), the four-point HMEG equation is attained as follows:

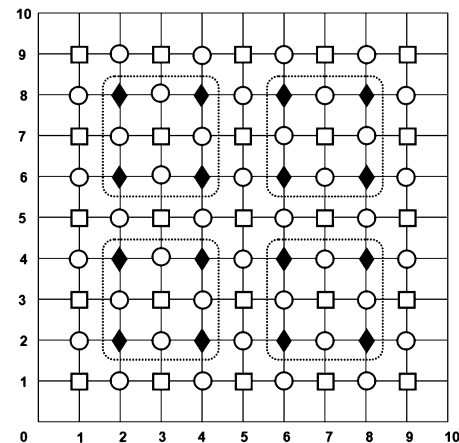
$$\begin{pmatrix} u_{ij}^{k+1} \\ u_{i+2,j}^{k+1} \\ u_{i+2,j+2}^{k+1} \\ u_{ij+2}^{k+1} \end{pmatrix} = \frac{1}{a^*} \begin{pmatrix} a_1^* & a_2^* & a_3^* & a_4^* \\ a_2^* & a_1^* & a_4^* & a_3^* \\ a_3^* & a_4^* & a_1^* & a_2^* \\ a_4^* & a_3^* & a_2^* & a_1^* \end{pmatrix} \begin{pmatrix} rhs_{ij} \\ rhs_{i+2,j} \\ rhs_{i+2,j+2} \\ rhs_{ij+2} \end{pmatrix}, \quad (14)$$

where

$$\begin{aligned}
 a^* &= (4 + d_1 + d_2 + 4\eta \Delta t)(4 + 3d_1 + d_2 + 4\eta \Delta t)(4 + d_1 + 3d_2 + 4\eta \Delta t) \\
 &\quad \times (4 + 3d_1 + 3d_2 + 4\eta \Delta t), \\
 a_1^* &= 8(2 + d_1 + d_2 + 2\eta \Delta t)(16 + 16d_1 + 3d_1^2 + 16d_2 + 8d_1d_2 + 3d_2^2 + 16d_1\eta \Delta t \\
 &\quad + 16d_2\eta \Delta t + 32\eta \Delta t + 16(\eta \Delta t)^2), \\
 a_2^* &= 4d_1(16 + 16d_1 + 3d_1^2 + 16d_2 + 8d_1d_2 + 5d_2^2 + 16d_1\eta \Delta t + 16d_2\eta \Delta t \\
 &\quad + 32\eta \Delta t + 16(\eta \Delta t)^2), \\
 a_3^* &= 16d_1d_2(2 + d_1 + d_2 + 2\eta \Delta t), \\
 a_4^* &= 4d_2(16 + 16d_1 + 5d_1^2 + 16d_2 + 8d_1d_2 + 3d_2^2 + 16d_1\eta \Delta t + 16d_2\eta \Delta t \\
 &\quad + 32\eta \Delta t + 16(\eta \Delta t)^2).
 \end{aligned}$$

In view of Fig. 2, all the grid points of the solution domain at any time level are partitioned into three distinct kinds of points ( $\blacklozenge$ ,  $\circ$ ,  $\square$ ). It can be observed that the implementation of Eq. (14) involves only points of kind  $\blacklozenge$ . Thus, we use Eq. (14) to iterate the solutions at these points until convergence is attained. After convergence is achieved, the HMEG method proceeds with the computations of the solutions at the residual points of kind  $\circ$  and  $\square$  directly once. For convenience, the four-point HMEG method is summarized in Algorithm 1.

**Figure 2** Four-point groups of the HMEG method with mesh size  $n = 10$




---

**Algorithm 1:** Solution algorithm using the HMEG iterative method

---

1. Partition the grid points of the solution domain into three kinds  $\blacklozenge$ ,  $\circ$  and  $\square$  as illustrated in Fig. 2.
2. Put all the  $\blacklozenge$  points into group of four points.
3. Use Eq. (14) to iterate the solutions at points of kind  $\blacklozenge$  in all the groups at time level  $k + 1$ .
4. Test the convergence. If the iterative solutions converge, move to step 5. Otherwise, repeat the iteration process at the same time level in step 3.
5. The solutions at the residual grid points  $\square$  and  $\circ$  are evaluated directly once as follows:
  - (a) For the points of kind  $\square$ , the rotated (skewed) finite difference scheme established from the skewed grid is utilized. This scheme is constructed by discretizing the approximating problem (5) on the skewed grid attained by rotating the standard grid  $45^\circ$  clockwise so that the following rotated difference scheme is obtained:

$$u_{ij}^{k+1} = \frac{1}{1 + \eta \Delta t + d_1 + d_2} \left[ \frac{d_1}{2} (u_{i+1,j-1}^{k+1} + u_{i-1,j+1}^{k+1}) + \frac{d_2}{2} (u_{i+1,j+1}^{k+1} + u_{i-1,j-1}^{k+1}) + u_{ij}^k + (r-1) \Delta t u_{ij}^0 + r \Delta t f_{ij}^{k+1} \right].$$

- (b) For the remaining points of kind  $\circ$ , the HSP formula (8) is employed.
- 

#### 4 Stability and convergence analyses

In this section, we will present the stability and convergence results. In view of the previous section, both the HEG and HMEG methods are derived from the same formula (8), but with different spacing. Thus, the stability and convergence analyses of both methods can be investigated in a similar manner. In the subsequent subsections, the matrix stability approach [28] together with mathematical induction will be used to analyze the stability and convergence of the HMEG method. Firstly, we recall the following remarks for the convenience of the subsequent analysis



**Remark 4.1** Let  $A_{n \times n}$  be an  $n \times n$  matrix. The infinity norm  $\|\cdot\|_\infty$  of the matrix  $A$  is given by

$$\|A_{n \times n}\|_\infty = \max_{1 \leq i \leq n} \left\{ \sum_{j=1}^n |a_{ij}| \right\}.$$

**Remark 4.2** ([33]) An  $n \times n$  matrix  $A_{n \times n}$  is said to be strictly diagonally dominant if  $|a_{ii}| > r_i(A)$ , where  $r_i(A) = \sum_{j \neq i, j=1}^n |a_{ij}|$ ,  $1 \leq i \leq n$  is the  $i$ th deleted absolute row sum.

**Remark 4.3** ([34]) If a matrix  $A_{n \times n}$  is strictly diagonally dominant, then  $A_{n \times n}$  is invertible and

$$\|A\|_\infty \leq \frac{1}{\min_{1 \leq i \leq n} \{|a_{ii}| - r_i(A)\}}.$$

#### 4.1 Stability analysis

Here, we analyze the stability of the HMEG method. For the sake of simplification, we assume that  $d_1 = d_2 = \Delta t/h^2$ . Consequently, Eq. (13) can be represented in matrix form as

$$Au^{k+1} = Bu^k + Cu^0 + b, \quad (15)$$

where

$$\begin{aligned} A &= \begin{pmatrix} J_1 & J_2 & & & \\ J_3 & J_1 & J_2 & & \\ & & \ddots & & \\ & & & J_3 & J_1 & J_2 \\ & & & J_3 & J_1 \end{pmatrix}, & B &= \begin{pmatrix} H & & & & \\ & H & & & \\ & & \ddots & & \\ & & & H & \\ & & & & H \end{pmatrix}, \\ C &= \begin{pmatrix} M & & & & \\ & M & & & \\ & & \ddots & & \\ & & & M & \\ & & & & M \end{pmatrix}, & b &= \begin{pmatrix} W_1 \\ W_1 \\ \vdots \\ W_1 \\ W_1 \end{pmatrix}, \\ J_1 &= \begin{pmatrix} Q_1 & Q_3 & & & \\ Q_2 & Q_1 & Q_3 & & \\ & & \ddots & & \\ & & & Q_2 & Q_1 & Q_3 \\ & & & Q_2 & Q_1 \end{pmatrix}, & J_2 &= \begin{pmatrix} Q_5 & & & & \\ & Q_5 & & & \\ & & \ddots & & \\ & & & Q_5 & \\ & & & & Q_5 \end{pmatrix}, \\ J_3 &= \begin{pmatrix} Q_4 & & & & \\ & Q_4 & & & \\ & & \ddots & & \\ & & & Q_4 & \\ & & & & Q_4 \end{pmatrix}, & H &= \begin{pmatrix} I_4 & & & & \\ & I_4 & & & \\ & & \ddots & & \\ & & & I_4 & \\ & & & & I_4 \end{pmatrix}, \end{aligned}$$

$$\begin{aligned}
M &= \begin{pmatrix} T_1 & & & \\ & T_1 & & \\ & & \ddots & \\ & & & T_1 \end{pmatrix}, & W_1 &= \begin{pmatrix} L_1 \\ L_1 \\ \vdots \\ L_1 \\ L_1 \end{pmatrix}, \\
Q_1 &= \begin{pmatrix} 1 + \eta\Delta t + d & -d/4 & 0 & -d/4 \\ -d/4 & 1 + \eta\Delta t + d & -d/4 & 0 \\ 0 & -d/4 & 1 + \eta\Delta t + d & -d/4 \\ -d/4 & 0 & -d/4 & 1 + \eta\Delta t + d \end{pmatrix}, \\
Q_2 &= \begin{pmatrix} 0 & 0 & 0 & -d/4 \\ 0 & 0 & -d/4 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, & Q_3 &= \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -d/4 & 0 & 0 \\ -d/4 & 0 & 0 & 0 \end{pmatrix}, \\
Q_4 &= \begin{pmatrix} 0 & -d/4 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -d/4 & 0 \end{pmatrix}, & Q_5 &= \begin{pmatrix} 0 & 0 & 0 & 0 \\ -d/4 & 0 & 0 & 0 \\ 0 & 0 & 0 & -d/4 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \\
T_1 &= \begin{pmatrix} (r-1)\Delta t & 0 & 0 & 0 \\ 0 & (r-1)\Delta t & 0 & 0 \\ 0 & 0 & (r-1)\Delta t & 0 \\ 0 & 0 & 0 & (r-1)\Delta t \end{pmatrix}, \\
I_4 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, & L_1 &= r\Delta t \begin{pmatrix} f_{i,j} \\ f_{i+2,j} \\ f_{i+2,j+2} \\ f_{i,j+2} \end{pmatrix}.
\end{aligned}$$

In the light of the distinguishing form of the matrices in Eq. (15), a further clarified form of Eq. (15) is found by writing

$$[A_{\frac{(n-2)^2}{4} \times \frac{(n-2)^2}{4}}]u^{k+1} = [B_{\frac{(n-2)^2}{4} \times \frac{(n-2)^2}{4}}]u^k + [C_{\frac{(n-2)^2}{4} \times \frac{(n-2)^2}{4}}]u^0 + b.$$

**Theorem 4.1** *The hybrid modified explicit group scheme (14) is unconditionally stable.*

*Proof* Suppose  $U^{k+1}$  is the approximate solution of (15). The error at time level  $k+1$  is defined as  $e^{k+1} = u^{k+1} - U^{k+1}$ . Considering Remarks 4.2 and 4.3, it follows that  $A$  is invertible and

$$u^{k+1} = A^{-1}Bu^k + A^{-1}Cu^0 + A^{-1}b. \quad (16)$$

From Eq. (16), the error satisfies

$$e^{k+1} = A^{-1}Be^k + A^{-1}Ce^0, \quad (17)$$

where

$$e^{k+1} = \begin{pmatrix} e_0^{k+1} \\ e_0^{k+1} \\ \vdots \\ e_0^{k+1} \end{pmatrix}, \quad e_0^{k+1} = \begin{pmatrix} \psi_1^{k+1} \\ \psi_2^{k+1} \\ \vdots \\ \psi_{\frac{(n-2)^2}{16}}^{k+1} \end{pmatrix}, \quad \psi^{k+1} = \begin{pmatrix} \psi_{ij}^{k+1} \\ \psi_{i+2,j}^{k+1} \\ \psi_{i+2,j+2}^{k+1} \\ \psi_{i,j+2}^{k+1} \end{pmatrix},$$

and  $\psi_{ij}^{k+1} = u_{ij}^{k+1} - U_{ij}^{k+1}$ .

To demonstrate the stability we shall prove that  $\|e^{k+1}\| \leq \|e^0\|$  for  $k = 0, 1, \dots, N-1$ . We use mathematical induction to prove it. For  $k = 0$ , we have

$$e^1 = A^{-1}Be^0 + A^{-1}Ce^0.$$

Since the matrix infinity norm  $\|A\|$  is consistent with the vector infinity norm  $\|e\|$ , we obtain

$$\begin{aligned} \|e^1\| &\leq \|A^{-1}B\| \|e^0\| + \|A^{-1}C\| \|e^0\| \\ &\leq \|A^{-1}\| \|B\| \|e^0\| + \|A^{-1}\| \|C\| \|e^0\|. \end{aligned}$$

As  $A$  is strictly diagonally dominant and using Remark 4.3, we have

$$\|e^1\| \leq \frac{1}{1+\eta\Delta t} \|e^0\| + \frac{(r-1)\Delta t}{1+\eta\Delta t} \|e^0\| = \frac{1+(r-1)\Delta t}{1+\eta\Delta t} \|e^0\| \leq \|e^0\|,$$

since  $r-1 < \eta$ ,

$$\therefore \|e^1\| \leq \|e^0\|.$$

Now, we assume that  $\|e^{s+1}\| \leq \|e^0\|$ ,  $s = 1, 2, \dots, k-1$ . We show this inequality is true for  $s = k$ .

Since  $r-1 < \eta$ , and from Eq. (17), we obtain

$$\begin{aligned} \|e^{k+1}\| &\leq \|A^{-1}\| \|B\| \|e^k\| + \|A^{-1}\| \|C\| \|e^0\| \\ &\leq \|A^{-1}\| \|B\| \|e^0\| + \|A^{-1}\| \|C\| \|e^0\| \\ &\leq \frac{1}{1+\eta\Delta t} \|e^0\| + \frac{(r-1)\Delta t}{1+\eta\Delta t} \|e^0\| \\ &= \frac{1+(r-1)\Delta t}{1+\eta\Delta t} \|e^0\| \leq \|e^0\|, \\ \therefore \|e^{k+1}\| &\leq \|e^0\|. \end{aligned}$$

This implies that the HMEG scheme (14) is unconditionally stable.  $\square$

## 4.2 Convergence analysis

Here, we follow an analogous approach as that in the previous subsection to investigate the convergence of the HMEG scheme (14).

**Theorem 4.2** *The hybrid modified explicit group scheme (14) is convergent and  $\|E^{k+1}\| \leq C_k(t + (\Delta x)^2 + (\Delta y)^2)$ .*

*Proof* Let  $R_{i,j}^{k+1}$  be the truncation error at the location  $(x_i, y_j, t_{k+1})$ . From Eq. (11), there is a positive constant  $C^*$  such that

$$|R_{i,j}^{k+1}| \leq C^*(t + (\Delta x)^2 + (\Delta y)^2), \quad (18)$$

where  $C^* = \max\{C_{i,j}^k\}$ ,  $i, j = 2, 3, \dots, n-2$ ,  $k = 0, 1, \dots, N-1$ .

We obtain the error equation by subtracting Eq. (15) from the following equation:

$$AU^{k+1} = BU^k + CU^0 + b + R^{k+1}.$$

The error equation immediately follows as

$$AE^{k+1} = BE^k + CE^0 + R^{k+1}, \quad (19)$$

where

$$E^{k+1} = \begin{pmatrix} E_0^{k+1} \\ E_0^{k+1} \\ \vdots \\ E_0^{k+1} \end{pmatrix}, \quad E_0^{k+1} = \begin{pmatrix} \phi_1^{k+1} \\ \phi_2^{k+1} \\ \vdots \\ \phi_{\frac{(n-2)^2}{16}}^{k+1} \end{pmatrix}, \quad \phi^{k+1} = \begin{pmatrix} \phi_{i,j}^{k+1} \\ \phi_{i+2,j}^{k+1} \\ \phi_{i+2,j+2}^{k+1} \\ \phi_{i,j+2}^{k+1} \end{pmatrix},$$

and  $\phi_{i,j}^{k+1} = U_{i,j}^{k+1} - u_{i,j}^{k+1}$ .

Next, we utilize mathematical induction to complete the proof. For  $k = 0$  and using that  $E^0 = 0$ , we have

$$AE^1 = R^1.$$

Then

$$\|E^1\| \leq \|A^{-1}\| \|R^1\| \leq \frac{1}{1 + \eta \Delta t} C^*(t + (\Delta x)^2 + (\Delta y)^2) = C_0(t + (\Delta x)^2 + (\Delta y)^2),$$

where  $C_0 = C^*/(1 + \eta \Delta t)$ .

$$\therefore \|E^1\| \leq C_0(t + (\Delta x)^2 + (\Delta y)^2).$$

Now, assume that  $\|E^{s+1}\| \leq C_s(t + (\Delta x)^2 + (\Delta y)^2)$ ,  $s = 1, 2, \dots, k-1$ . We show this inequality is true for  $s = k$ .

From Equation (19), we obtain

$$\begin{aligned} \|E^{k+1}\| &\leq \|A^{-1}\| \|B\| \|E^k\| + \|A^{-1}\| \|R^{k+1}\| \\ &\leq \frac{1}{1 + \eta \Delta t} \|E^k\| + \frac{1}{1 + \eta \Delta t} \|R^{k+1}\| \end{aligned}$$

$$\begin{aligned} &\leq \frac{1}{1 + \eta \Delta t} [C_{k-1}(t + (\Delta x)^2 + (\Delta y)^2) + C^*(t + (\Delta x)^2 + (\Delta y)^2)] \\ &= C_k(t + (\Delta x)^2 + (\Delta y)^2), \end{aligned}$$

where  $C_k = C_{k-1} + C^*$  since  $\lim_{k \rightarrow \infty} \Delta t = 0$ .

$$\therefore \|E^{k+1}\| \leq C_k(t + (\Delta x)^2 + (\Delta y)^2).$$

Hence, the proof is completed.  $\square$

## 5 Numerical experiments and results

In this part, we carry out computer simulations to investigate the performance of the hybrid group iterative methods developed in this work, and to compare their performances with the HSP iterative method which was developed in [17]. The computational experiments were conducted in Mathematica software and run on a laptop with quad core processor, 8 GB of RAM and Windows 10 operating system. In practice, the Gauss–Seidel method with a fixed relaxation factor of 1 was employed to obtain the numerical results. For convenience, the  $l_\infty$  norm along with a tolerance factor of  $10^{-5}$  were utilized for the convergence criteria throughout the computational experiments.

In developing fast iterative numerical schemes, the computational cost estimated by the total number of arithmetic operations to be implemented per iteration is a crucial determinant. The higher the number of arithmetic operations to be executed (i.e. higher computational cost), the more the algorithm's computational time, and hence slowness in the convergence is indicated. Here, the computational cost of the presented methods is measured by computing the total arithmetic operations involved for each method as illustrated in Table 1. For further details about the computational cost of the group iterative schemes, kindly refer to [25, 28].

In order to illustrate the validity of the proposed methods, the maximum error norm is applied using the following formula:

$$Error_\infty = \max_{i,j} |u_{i,j}^{\text{exact}} - u_{i,j}^{\text{num}}|.$$

**Example 5.1** In this example, we specify a solution domain of  $\Omega = \{(x, y, t) | 0 \leq x, y \leq 1, 0 \leq t \leq 1\}$  for solving the following two-dimensional time-fractional cable equation [35]:

$${}^C D_t^\alpha u(x, y, t) = \frac{\partial^2 u(x, y, t)}{\partial x^2} + \frac{\partial^2 u(x, y, t)}{\partial y^2} - u(x, y, t) + \left( \frac{2t^{2-\alpha}}{\Gamma(3-\alpha)} - t^2 \right) e^{x+y},$$

with the exact solution given by  $u(x, y, t) = t^2 e^{x+y}$ .

**Table 1** The computational cost of the HSP, HEG and HMEG methods ( $\sigma = n - 1$ )

Method	Per iteration	After convergence	Total operations
HSP	$21\sigma^2 * lte$	–	$21\sigma^2 * lte$
HEG	$(19(\sigma - 1)^2 + 21(2\sigma - 1)) * lte$	–	$(19(\sigma - 1)^2 + 21(2\sigma - 1)) * lte$
HMEG	$5.25(\sigma - 1)^2 * lte$	$5.25(3\sigma^2 + 2\sigma - 1)$	$5.25(\sigma - 1)^2 * lte + 5.25(3\sigma^2 + 2\sigma - 1)$

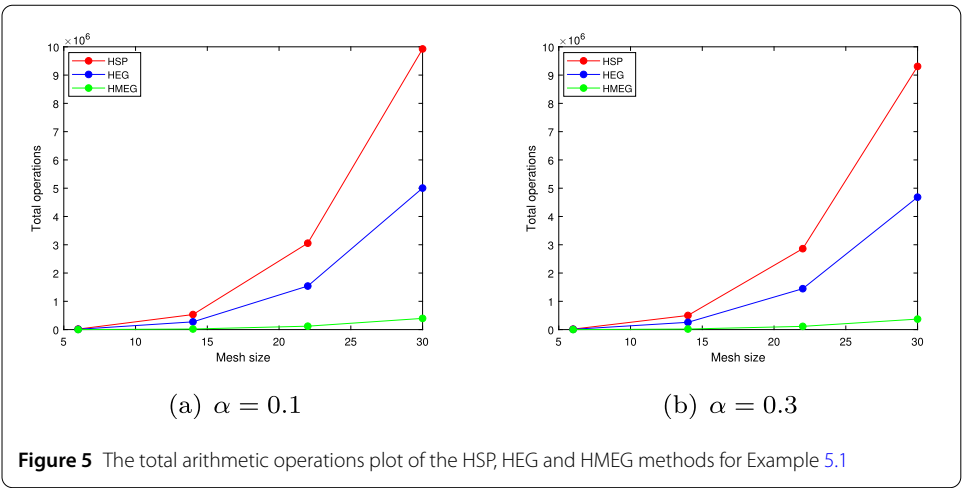
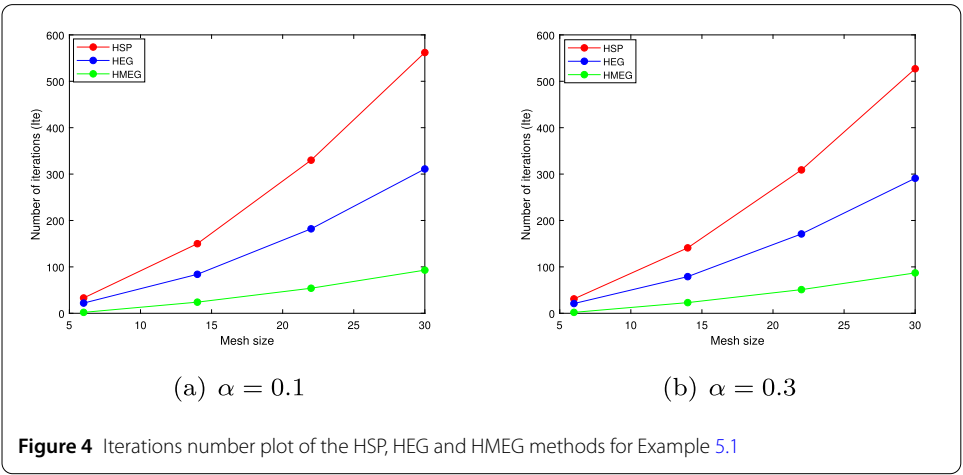
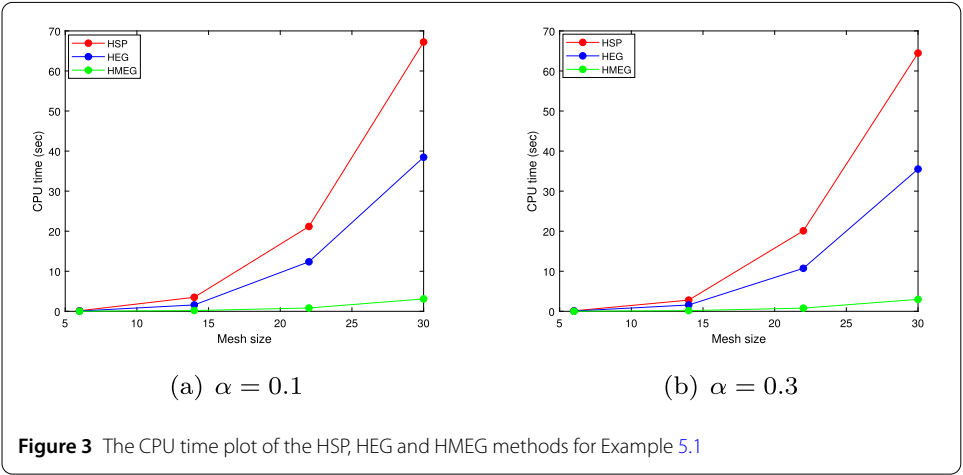
**Table 2** The numerical results of the HSP, HEG and HMEG methods at  $\alpha = 0.1$  for Example 5.1

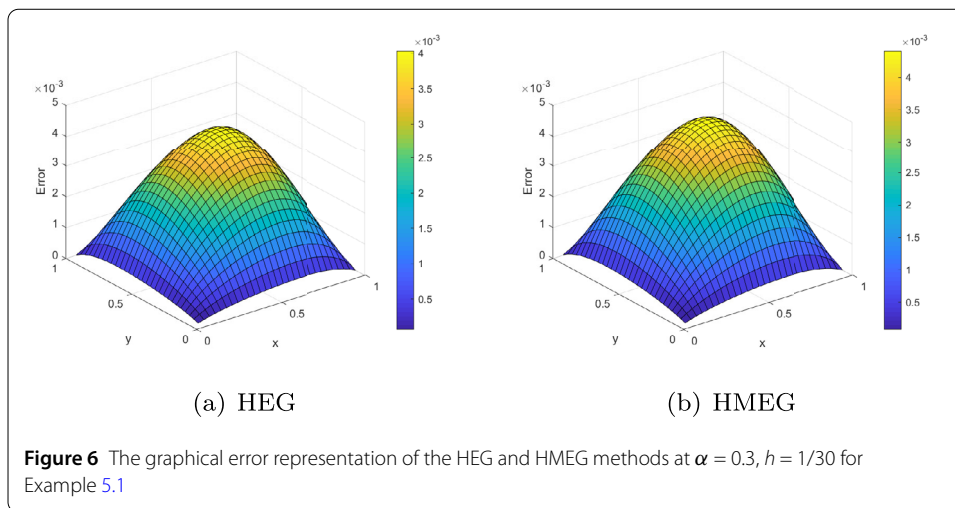
$h^{-1}$	Method	CPU time (sec)	<i>Ite</i>	$Error_{\infty}$	Total operations
6	HSP	0.140	33	1.5652E-03	17,325
	HEG	0.078	22	1.5795E-03	10,846
	HMEG	0.015	2	3.9024E-03	609
14	HSP	3.500	150	8.2547E-04	532,350
	HEG	1.593	84	8.9366E-04	273,924
	HMEG	0.187	24	1.4212E-03	20,937
22	HSP	21.171	330	5.4014E-04	3,056,130
	HEG	12.359	182	6.9130E-04	1,539,902
	HMEG	0.828	54	1.0299E-03	120,561
30	HSP	67.234	562	3.1519E-04	9,925,482
	HEG	38.468	311	5.2364E-04	5,004,923
	HMEG	3.093	93	8.7262E-04	396,333

**Table 3** The numerical results of the HSP, HEG and HMEG methods at  $\alpha = 0.3$  for Example 5.1

$h^{-1}$	Method	CPU time (sec)	<i>Ite</i>	$Error_{\infty}$	Total operations
6	HSP	0.125	31	4.8575E-03	16,275
	HEG	0.078	21	4.8648E-03	10,353
	HMEG	0.015	2	6.9708E-03	609
14	HSP	2.796	141	4.3450E-03	500,409
	HEG	1.578	79	4.4206E-03	257,619
	HMEG	0.171	23	4.8858E-03	20,181
22	HSP	20.093	309	4.0362E-03	2,861,649
	HEG	10.734	171	4.2367E-03	1,446,831
	HMEG	0.796	51	4.5519E-03	114,261
30	HSP	64.468	527	3.6839E-03	9,307,347
	HEG	35.515	291	4.0316E-03	4,683,063
	HMEG	2.984	87	4.4048E-03	371,637

The initial and boundary conditions of this problem are derived from the above exact solution. In solving this problem, several mesh sizes of 6, 14, 22 and 30 have been utilized for the space discretization with fixed temporal step size of  $\Delta t = 1/10$ . The obtained results of the CPU computational time (in seconds), number of iterations (*Ite*), total number of arithmetic operations (Total operations) and numerical errors ( $Error_{\infty}$ ) for the presented methods described in Sects. 2 and 3 are compared in Tables 2 and 3 when  $\alpha = 0.1$  and 0.3, respectively. Clearly, it can be seen that the proposed hybrid group iterative methods are able to reduce the iterations number, computational cost and hence the CPU time significantly compared to the HSP iterative method [17], without deteriorating the accuracy of numerical solutions. From the experimental results, the CPU time, iterations number and total arithmetic operations of the HEG method are, respectively, only about 45.51–62.40%, 55.15–67.47% and 50.31–63.61% of the HSP method. Similarly, the CPU time, iterations number and total operations of the HMEG method are, respectively, only about 3.91–12.00%, 6.06–16.54% and 3.51–3.99% of the HSP method. The comparison of the computational results for the hybrid iterative methods are illustrated in Figs. 3, 4 and 5. Figure 6 depicts the graphical error representation of the HEG and HMEG methods when  $\alpha = 0.3$ . In view of this figure, the hybrid group iterative methods are able to simulate Example 5.1 precisely and rather quickly.



**Table 4** The numerical results of the HSP, HEG and HMEG methods at  $\alpha = 0.7$  for Example 5.2

$h^{-1}$	Method	CPU time (sec)	$l_{te}$	$Error_{\infty}$	Total operations
6	HSP	0.109	24	2.4056E-02	12,600
	HEG	0.062	16	2.4069E-02	7888
	HMEG	0.015	2	8.8093E-02	609
22	HSP	14.187	219	4.7123E-03	2,028,159
	HEG	7.796	123	4.9438E-03	1,040,703
	HMEG	0.671	38	9.6396E-03	86,961
38	HSP	102.313	540	2.8059E-03	15,524,460
	HEG	60.765	307	3.4906E-03	8,030,199
	HMEG	5.156	96	5.5162E-03	675,129
54	HSP	379.203	944	1.1587E-03	55,685,616
	HEG	215.391	545	2.5382E-03	29,201,645
	HMEG	19.765	174	4.3268E-03	2,514,897

**Example 5.2** Here, we take the following two-dimensional cable equation of fractional order [4]:

$$\begin{aligned}
 {}^C_0 D_t^\alpha u(x, y, t) &= \frac{\partial^2 u(x, y, t)}{\partial x^2} + \frac{\partial^2 u(x, y, t)}{\partial y^2} - u(x, y, t) \\
 &\quad + \left( \frac{2t^{2-\alpha}}{\Gamma(3-\alpha)} + (1 + 2\pi^2)t^2 \right) \sin(\pi x) \sin(\pi y),
 \end{aligned}$$

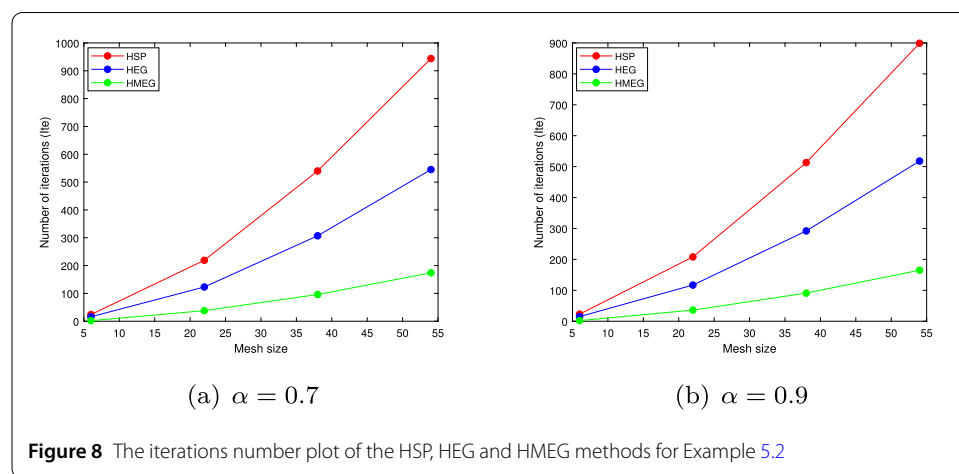
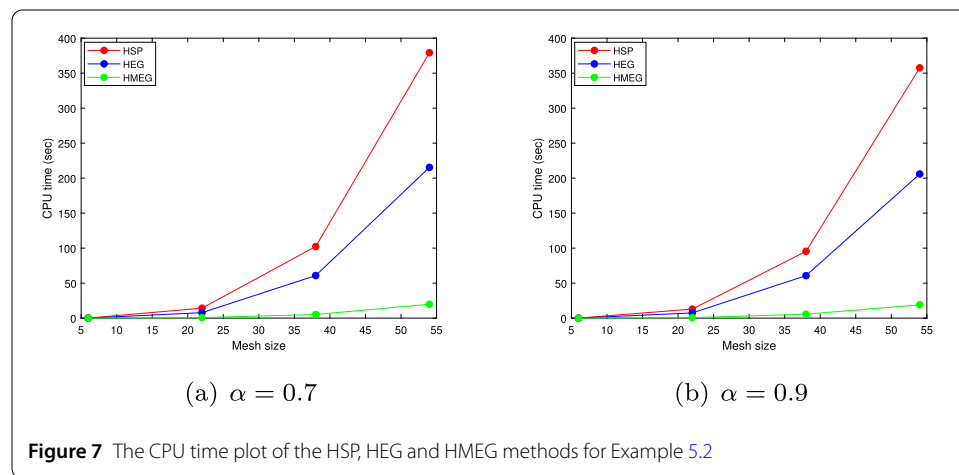
subject to the initial and boundary conditions extracted from the exact solution  $u(x, y, t) = t^2 \sin(\pi x) \sin(\pi y)$ .

For the solution of this problem, we determine the solution domain as  $\Omega = \{(x, y, t) | 0 \leq x, y \leq 1, 0 \leq t \leq 1\}$ . Various mesh sizes of 6, 22, 38 and 54 and fixed temporal step size  $\Delta t = 1/10$  are utilized to discretize the solution domain. Tables 4 and 5 summarize the numerical results obtained by using the HSP, HEG and HMEG methods when  $\alpha = 0.7$  and 0.9, respectively. From the computational results, the CPU time, iterations number and total arithmetic operations of the HEG method are, respectively, only about 49.46–63.56%, 56.16–66.66% and 51.31–62.60% of the HSP method. On the other hand, the CPU time, iterations number and total operations of the HMEG method are, respectively, only about 4.72–16.12%, 8.33–18.43% and 4.28–5.04% of the HSP method. In Figs. 7, 8 and 9 we

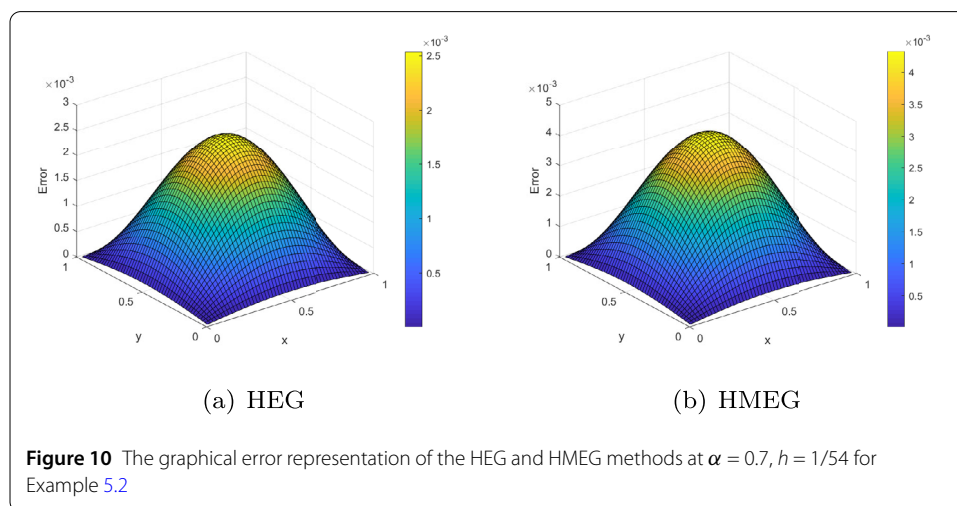
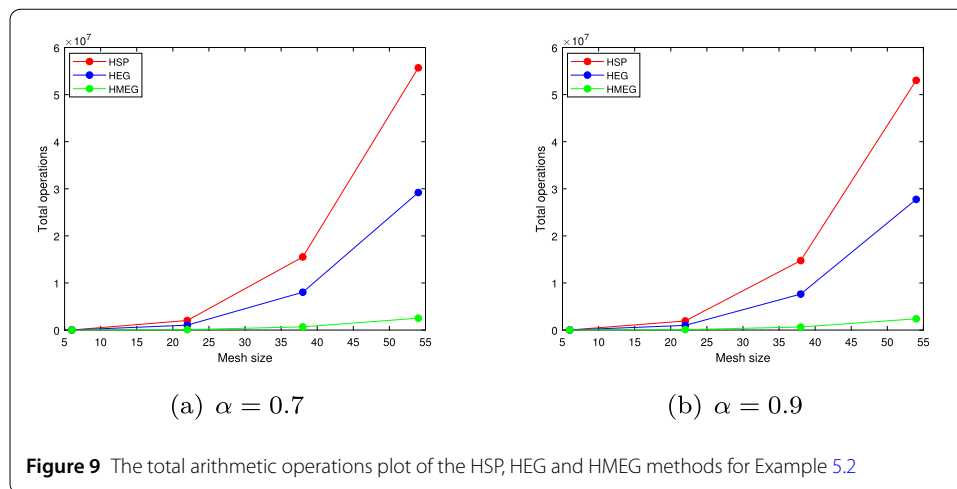


**Table 5** The numerical results of the HSP, HEG and HMEG methods at  $\alpha = 0.9$  for Example 5.2

$h^{-1}$	Method	CPU time (sec)	$l_{te}$	$Error_{\infty}$	Total operations
6	HSP	0.093	23	2.4887E-02	12,075
	HEG	0.046	15	2.4897E-02	7395
	HMEG	0.015	2	8.8361E-02	609
22	HSP	12.859	208	5.6912E-03	1,926,288
	HEG	7.406	117	5.9235E-03	989,937
	HMEG	0.625	36	1.0583E-02	82,761
38	HSP	95.453	513	3.7797E-03	14,748,237
	HEG	60.671	292	4.4777E-03	7,637,844
	HMEG	5.625	91	6.9404E-03	641,109
54	HSP	357.594	899	2.1237E-03	53,031,111
	HEG	205.953	518	3.5163E-03	27,754,958
	HMEG	18.984	165	5.3069E-03	2,387,133



sketch the CPU time, iterations number and total operations of the presented methods by fixing all the parameters and only altering the mesh size. In each figure, the computational outcomes of the HEG and HMEG methods are considerably less than those of the HSP method, whereas the HMEG method has the least computing effort among these methods. This is in good agreement with the theoretical computational cost analysis. Figure 10 displays the graphical error representation using the HEG and HMEG methods



when  $\alpha = 0.7$ . It can be observed that the proposed hybrid group iterative methods are computationally efficient in the sense that they could obtain a satisfying error with rather least computational cost and CPU time.

## 6 Conclusions

In this article, two hybrid group iterative methods based on the Laplace transform method and group iterative schemes have been proposed for solving the two-dimensional time-fractional cable equation. The HEG method is formulated from the  $h$ -spaced implicit finite difference scheme, whereas the HMEG method is derived from the  $2h$ -spaced implicit finite difference approximation. The unconditional stability and convergence of the HMEG method is proved using matrix stability approach. The computational cost (arithmetic operations per iteration) of the presented methods has been analyzed and verified with the help of examples. Numerical experiments strongly support theoretical analyses and illustrate the computational efficiency of the proposed methods. The corresponding numerical results show that the hybrid group iterative methods could simulate the problem precisely and reduce the computational cost, iterations number as well as CPU time significantly when compared to the HSP iterative method [17], where the least computing

effort has shown to be required by the HMEG method. The development of hybrid group iterative methods together with the corresponding theoretical analyses will be considered in future work.

#### Acknowledgements

The authors extend their sincere appreciation to the editor and referees for their time and valuable comments. The authors also gratefully acknowledge the financial support from Universiti Sains Malaysia (USM) Research University Grant (1001/PMATHS/8011101).

#### Funding

This research was funded by the Universiti Sains Malaysia (USM), School of Mathematical Sciences.

#### Availability of data and materials

Data sharing not applicable to this article as no data sets were generated or analysed during the current study.

#### Competing interests

The authors declare that they have no competing interests.

#### Authors' contributions

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

#### Publisher's Note

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

Received: 2 March 2020 Accepted: 21 May 2020 Published online: 02 June 2020

#### References

1. Sweilam, N.H., Khader, M.M., Adel, M.: Numerical simulation of fractional cable equation of spiny neuronal dendrites. *J. Adv. Res.* **5**, 253–259 (2014)
2. Hu, X., Zhang, L.: Implicit compact difference schemes for the fractional cable equation. *Appl. Math. Model.* **36**, 4027–4043 (2012)
3. Zhang, H., Yang, X., Han, X.: Discrete-time orthogonal spline collocation method with application to two-dimensional fractional cable equation. *Comput. Math. Appl.* **68**, 1710–1722 (2014)
4. Liu, J., Li, H., Liu, Y.: A new fully discrete finite difference/element approximation for fractional cable equation. *J. Appl. Math. Comput.* **52**, 345–361 (2016)
5. Yu, B., Jiang, X.: Numerical identification of the fractional derivatives in the two-dimensional fractional cable equation. *J. Sci. Comput.* **68**, 252–272 (2016)
6. Li, M.Z., Chen, L.J., Xu, Q., Ding, X.H.: An efficient numerical algorithm for solving the two-dimensional fractional cable equation. *Adv. Differ. Equ.* **2018**, 424 (2018)
7. Liu, Z., Cheng, A., Li, X.: A fast-high order compact difference method for the fractional cable equation. *Numer. Methods Partial Differ. Equ.* **34**, 2237–2266 (2018)
8. Sweilam, N.H., Al-Mekhlafi, S.M.: A novel numerical method for solving the 2-D time fractional cable equation. *Eur. Phys. J. Plus* **134**, 323 (2019)
9. Li, X., Rui, H.: Stability and convergence based on the finite difference method for the nonlinear fractional cable equation on non-uniform staggered grids. *Appl. Numer. Math.* **152**, 403–421 (2020)
10. Atangana, A., Owolabi, K.M.: New numerical approach for fractional differential equations. *Math. Model. Nat. Phenom.* **13**, 3 (2018)
11. Akgül, A., Modanli, M.: Crank–Nicolson difference method and reproducing kernel function for third order fractional differential equations in the sense of Atangana–Baleanu Caputo derivative. *Chaos Solitons Fractals* **127**, 10–16 (2019)
12. Atangana, A., Akgül, A., Owolabi, K.M.: Analysis of fractal fractional differential equations. *Alex. Eng. J.* (2020). <https://doi.org/10.1016/j.aej.2020.01.005>
13. Atangana, A.: Fractional discretization: the African's tortoise walk. *Chaos Solitons Fractals* **130**, 109399 (2020)
14. Gong, C., Bao, W., Tang, G., Jiang, Y., Liu, J.: Computational challenge of fractional differential equations and the potential solutions: a survey. *Math. Probl. Eng.* **2015**, Article ID 258265 (2015)
15. Jiang, S., Zhang, J., Zhang, Q., Zhang, Z.: Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations. *Commun. Comput. Phys.* **21**, 650–678 (2017)
16. Salama, F.M., Ali, N.H.M.: Computationally efficient hybrid method for the numerical solution of the 2D time fractional advection–diffusion equation. *Int. J. Math. Eng. Manag. Sci.* **5**, 432–446 (2020)
17. Salama, F.M., Ali, N.H.M.: Fast  $O(N)$  hybrid method for the solution of two dimensional time fractional cable equation. *Comput. Softw.* **8**, 3453–3461 (2019)
18. Smith, G.D.: *Numerical Solution of Partial Differential Equations: Finite Difference Methods*. Oxford University Press, Oxford (1985)
19. Yousif, W.S., Evans, D.J.: Explicit group over-relaxation methods for solving elliptic partial differential equations. *Math. Comput. Simul.* **28**, 453–466 (1986)
20. Evans, D.J., Sahimi, M.S.: The alternating group explicit (AGE) iterative method for solving parabolic equations I: 2-dimensional problems. *Int. J. Comput. Math.* **24**, 311–341 (1988)
21. Evans, D.J., Yousif, W.S.: Explicit group iterative methods for solving elliptic partial differential equations in 3-space dimensions. *Int. J. Comput. Math.* **18**, 323–340 (1986)

22. Abdullah, A.R.: The four point explicit decoupled group (EDG) method: a fast Poisson solver. *Int. J. Comput. Math.* **38**, 61–70 (1991)
23. Yousif, W.S., Evans, D.J.: Explicit de-coupled group iterative methods and their parallel implementations. *Parallel Algorithms Appl.* **7**, 53–71 (1991)
24. Othman, M., Abdullah, A.R.: An efficient four points modified explicit group Poisson solver. *Int. J. Comput. Math.* **76**, 203–217 (2000)
25. 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)
26. 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)
27. Evans, D.J., Biggins, M.J.: The solution of elliptic partial differential equations by a new block over-relaxation technique. *Int. J. Comput. Math.* **10**, 269–282 (1982)
28. Balasim, A.T., HJ. Mohd. Ali, N.: New group iterative schemes in the numerical solution of the two-dimensional time fractional advection–diffusion equation. *Cogent Math.* **4**, 1412241 (2017)
29. Ali, A., Ali, N.H.M.: Explicit group iterative methods in the solution of two dimensional time-fractional diffusion-wave equation. *Composoft* **7**, 2931–2938 (2018)
30. Rahman, R., Ali, N.A.M., Sulaiman, J., Muhiddin, F.A.: Block iterative method for the solution of fractional two-point boundary value problems. *J. Phys. Conf. Ser.* **1358**, 012053 (2019)
31. Atangana, A., Akgül, A.: Can transfer function and Bode diagram be obtained from Sumudu transform. *Alex. Eng. J.* (2020). <https://doi.org/10.1016/j.aej.2019.12.028>
32. Ren, J., Sun, Z.Z., Dai, W.: New approximations for solving the Caputo-type fractional partial differential equations. *Appl. Math. Model.* **40**, 2625–2636 (2016)
33. Horn, R.A., Johnson, C.R.: *Topics in Matrix Analysis*. Cambridge University Press, Cambridge (1991)
34. Morača, N.: Bounds for norms of the matrix inverse and the smallest singular value. *Linear Algebra Appl.* **429**, 2589–2601 (2008)
35. Bhrawy, A.H., Zaky, M.A.: Numerical simulation for two-dimensional variable-order fractional nonlinear cable equation. *Nonlinear Dyn.* **80**, 101–116 (2015)

**Submit your manuscript to a SpringerOpen<sup>®</sup> 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)