Networks and Heterogeneous Media

https://www.aimspress.com/journal/nhm

NHM, 20(3): 818–843.
DOI: 10.3934/nhm.2025035
Received: 1 April 2025
Revised: 22 May 2025
Accepted: 23 June 2025
Published: 16 July 2025

Research article

The space-time generalized finite difference method for the time fractional mobile/immobile diffusion equation

Haili Qiao*

School of Mathematical Sciences, Liaocheng University, Liaocheng, Shandong, China

* **Correspondence:** Email: QHLmath@163.com.

Abstract: The space-time generalized finite difference method was used to solve the time fractional mobile/immobile diffusion equation. Traditionally, partial differential equations are solved by separating time and space dimensions, and applying distinct numerical methods to each. Here, the time dimension was treated as an additional spatial dimension, thus transforming the d-dimensional spatial problem into a new $(d+1)$-dimensional problem in the space-time domain. In the space-time generalized finite difference method, the complexity of the numerical solution of the problem was effectively reduced by simultaneously discretizing the space and time dimensions in the space-time domain, while retaining all the advantages of the generalized finite difference method. We took the second-order Taylor series expansion as an example to show the numerical algorithm of the 2D time fractional mobile/immobile diffusion equation, and numerically simulated the one- and two-dimensional problems where the solution was a sufficiently smooth or weak singularity at the initial moment, thereby verifying the effectiveness of the algorithm.

Keywords: time fractional mobile/immobile diffusion equation; the space-time generalized finite difference method; numerical examples

1. Introduction

Consider the following time fractional mobile/immobile diffusion problem:

$$\begin{aligned} &\frac{\partial u}{\partial t}(\mathbf{x},t) + {}_0^C D_t^\alpha u(\mathbf{x},t) = \triangle u(\mathbf{x},t) + f(\mathbf{x},t), \quad (\mathbf{x},t) \in \Omega \times (0,T], \\ &a_i u(\mathbf{x},t) + b_i \bigtriangledown u(\mathbf{x},t) \cdot \mathbf{n} = \phi_i(\mathbf{x},t), \quad \mathbf{x} \in \partial\Omega_i, \quad t \in (0,T], \\ &u(\mathbf{x},0) = u_0(\mathbf{x}), \quad \mathbf{x} \in \overline{\Omega}, \end{aligned} \tag{1.1}$$

where $0 < \alpha < 1$, $\triangle$ represents the Laplacian operator, $\Omega = (0,L)^d \subset R^d$, $d = 1,2$, with boundary $\partial\Omega$, $\cup_i \partial\Omega_i = \partial\Omega$, when $a_i \neq 0,\ b_i = 0$, $\partial\Omega_i = \partial\Omega$, the problem (1.1) has Dirichlet boundary conditions, $\overline{\Omega} =$

$\Omega \bigcup \partial\Omega$, and $\mathbf{n}$ represents the unit normal vector . Assuming that $f(\mathbf{x}, t)$ satisfies a certain smoothness, and $u_0(\mathbf{x})$ is a given continuous function, ${}_0^C D_t^\alpha u(\mathbf{x}, t)$ denotes the Caputo fractional derivative for the time variable t, defined as follows:

$$ {}_0^C D_t^\alpha u(\mathbf{x}, t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t (t-s)^{-\alpha} \frac{\partial u(\mathbf{x}, s)}{\partial s} ds. \tag{1.2} $$

The fractional mobile/immobile transport model was initially proposed by Schumer et al. [1], which introduces a time derivative term to describe the motion time, thereby distinguishing between the moving and stationary states of particles (see [2, 3]). The model also describes problems such as ocean acoustic propagation and thermal diffusion, has important applications in groundwater solute modeling [4, 5], and can also be considered as the limiting equation controlling continuous time stochastic walks with heavy-tailed random waiting times [1]. Therefore, the fractional mobile/immobile transport model holds significant research importance, and many researchers have attempted to propose stable and efficient numerical algotithms to obtain reliable approximate solutions.

So far, various methods have been proposed to simulate the time fractional mobile/immobile equations. Zheng and Wang [6] proposed the averaged L1 compact finite difference method for solving time fractional mobile/immobile diffusion equations with weakly singular solutions. This method is unconditionally convergent with a convergence order of $O(\tau^2 \ln(\tau) + h^4)$. Fardi and Ghasemi [7] proposed a finite difference/spectral method based on Caputo fractional derivatives for solving time fractional mobile/immobile transport models, and demonstrated the stability and convergence of the method. Zhang et al. [8] proposed a weighted and shifted Grünwald-Letnikov difference/Legendre spectral method for solving two dimensional nonlinear time fractional mobile/immobile convection diffusion equations. Their work introduced correction techniques to address time singularities and provided theoretical analysis of the method. Nikan et al. [9] developed an efficient algorithm based on the radial basis function finite difference method for solving time fractional mobile/immobile transport models. The time fractional derivative is discretized by the Grünwald-Letnikov difference, and the stability and convergence of the numerical scheme are rigorously demonstrated through the energy method. Zhao et al. [10] introduced a finite volume element method with a second-order weighted and shifted Grünwald difference scheme for solving nonlinear time fractional mobile/immobile transport models, and proved unconditional stability of the fully discrete scheme. Jiang et al. [11] proposed an alternating direction implicit (ADI) compact difference scheme for solving two-dimensional semilinear time fractional mobile/immobile equations. The time first derivative and Caputo derivative are discretized using the second-order backward differential scheme and L1 scheme, respectively, and the stability and convergence are analyzed using the energy method. Qiu et al. [12] proposed a time two grid algorithm based on the finite difference method for solving two-dimensional nonlinear time fractional mobile/immobile transport models. The time derivative is discretized by the second-order backward difference formula, the Caputo fractional derivative is discretized by the L1 formula, and the spatial derivative is approximated by the central difference formula. The stability and convergence of the two grid finite difference scheme are also proved. However, previous studies have typically discretized space and time using different numerical methods and then combined these methods, making it difficult to define the stability of the numerical schemes. In addition, existing algorithms mainly solve problem (1.1) with regular domains and Dirichlet boundary conditions, which are difficult to apply in practice. In this paper, we consider

using the space-time generalized finite difference method to investigate the time fractional mobile/immobile equation.

The generalized finite difference method (GFDM) is one of the most promising meshless methods developed from Taylor series expansion and the weighted moving least squares method, which does not require mesh construction of the numerical product, and its core idea is to represent the derivative of the unknown function by linear combination of function values at neighboring points. Recently, the generalized finite difference method has been widely used to solve various scientific and engineering problems. Qin et al. [13] proposed a meshless method for solving three-dimensional elliptical interface problems based on the generalized finite difference method. The accuracy and stability of the method were verified through numerical examples, and some examples showed that the method is more accurate than the classical immersion finite element method. Gu et al. [14] proposed a new framework for solving three-dimensional dynamic coupled thermoelastic problems, introducing the Krylov-Delay correction method for large-scale and long-time simulations. Then, the generalized finite difference method was used to solve the resulting boundary value problem, and a new distance standard for adaptive node selection in the generalized finite difference method simulations was studied. Gu et al. [15] proposed a meshless numerical method based on the generalized finite difference method for recovering time-dependent heat sources in three-dimensional heat conduction problems, and verified the accuracy and efficiency of the proposed method through numerical examples. Song et al. [16] used the generalized finite difference method to solve the steady-state two-dimensional and three-dimensional Stokes equations, and numerical examples verified the feasibility of the generalized finite difference method. Ureña et al. [17] demonstrated the application of the generalized finite difference method in solving various nonlinear problems, including heat transfer, acoustics, and mass transfer problems. Rao et al. [18] developed a new meshless method called the node control domain meshless method for dealing with porous flow problems with singular source terms by virtually constructing node control domains. In theory, the proposed node control domain meshless method has the advantages of previous meshless methods for discretizing computational domains with complex geometries, as well as the advantages of the traditional low-order finite volume method for stabilizing various porous flow problems with local mass conservation. Zhan et al. [19] applied the generalized finite difference method to compressible two-phase flow in anisotropic porous media, revealing the enormous potential of the generalized finite difference method in numerical simulation of oil and gas reservoirs. Rao et al. [20] applied the newly developed the upwind generalized finite difference method to the two-phase porous flow equation. The computational performance of upwind generalized finite difference method was analyzed, providing important reference for the development of general meshless numerical methods for porous flow problems. These problems are all integer-order partial differential equations, and some researchers have applied the generalized finite difference method to fractional partial differential equations. Sun et al. [21] used the generalized finite difference method to solve a class of multidimensional fractional diffusion equations in finite fields, and verified the effectiveness of the algorithm through numerical examples. Hu et al. [22] used the generalized finite difference method to handle the inverse Cauchy problem related to fractional heat conduction models with functionally graded materials. Time discretization was achieved using the standard implicit finite difference method, and spatial discretization was achieved through the generalized finite difference method, which effectively avoided the ill posedness in the inverse Cauchy problem. Hosseini et al. [23]

implemented the generalized finite difference method to solve reaction-diffusion equations. Various computational domains were considered in the numerical simulation, demonstrating the reliability and accuracy of the proposed method. The sensitivity of selecting local matching points was also reported.

The space-time generalized finite difference method (ST-GFDM) effectively avoids the semi-discretization problem in the space-time domain by combining the space-time coupling approach with the generalized finite difference method. This method prevents the accumulation of numerical errors at each time layer in numerical simulations, while preserving the advantages of the generalized finite difference method. Additionally, it enhances computational efficiency and reduces the complexity of numerical solutions. In the space-time generalized finite difference method, the derivative of an unknown variable at a point is expressed by the linear combination of the function values at nearby nodes. Qu and He [24] proposed a space-time generalized finite difference method with additional conditions for two-dimensional transient heat conduction analysis of functionally graded materials. The performance of the method was demonstrated through numerical experiments, and the numerical accuracy of this method was compared with the traditional generalized finite difference method. Li [25] combined the space-time generalized finite difference method with Newton's method to solve the two-dimensional unsteady Burger equation. The accuracy of the algorithm is demonstrated through numerical examples. Liu et al. [26] used the space-time generalized finite difference method with auxiliary nodes to solve the bending problem of elastic thin plates under dynamic loads, and demonstrated the accuracy and stability of the algorithm through several numerical examples. Zhang et al. [27] applied the space-time generalized finite difference scheme to solve the multidimensional nonlinear high-order Korteweg-de Vries equation. The proposed numerical scheme was obtained by combining the space-time generalized finite difference method, Levenberg Marquardt algorithm, and time marching method. In the calculation process, the time marching method reduced the computational workload and improved efficiency by dividing the space-time domain. Benito et al. [28] applied the space-time cloud method and generalized finite difference method to solve the Zakharov Kuznetsov modified equal width equation, proved the convergence of the explicit generalized finite difference method, and compared the accuracy and computational efficiency of the two methods. Li et al. [29] applied the space-time generalized finite difference scheme to solve nonlinear dispersive shallow water waves. The proposed meshless numerical scheme was obtained by combining the space-time generalized finite difference method, two-step Newton method, and time marching method. Numerical examples verified the effectiveness of the algorithm. Benito et al. [30] applied the space-time generalized finite difference method to solve partial differential equations, numerically simulated 2D and 3D examples of regular and irregular node distributions, and compared the results with the explicit generalized finite difference method in terms of error and execution time. Li et al. [31] combined the space-time generalized finite difference method with the Newton Raphson and time-marching methods to stably solve the nonlinear equal width equation in long-time simulation. The matrix obtained by numerical discretization was effectively solved using the Newton Raphson method. In addition, the time-marching method was used to process the ST domain along the time axis. As far as I know, the space-time generalized finite difference method has only been used for integer-order partial differential equations at present, and has not been applied to time-fractional partial differential equations.

The rest of this paper is organized as follows. In Section 2, the space-time generalized finite difference method based on the second-order Taylor series expansion is introduced. In Section 3, the

time fractional mobile/immobile diffusion Eq (1.1) is discretized. In Section 4, numerical simulations are conducted for various examples to verify the effectiveness of the algorithm. In Section 5, the algorithm is discussed. Finally, in Section 6, the paper is summarized.

2. The space-time generalized finite difference method

This section briefly introduces the space-time generalized finite difference method.

In the space-time generalized finite difference method, the time direction is treated as an additional dimension, allowing a time-dependent d-dimensional problem to be considered as a $(d+1)$-dimensional steady-state problem, and the numerical procedure of the generalized finite difference method can be directly used for the $(d+1)$-dimensional problem. For illustrative purposes, this paper focuses on the time-dependent 2D problem in the $x-y-t$ space-time domain. The generalized finite difference method employs the second-order Taylor series expansion. After dispersing the points over the entire space-time computational domain, each point has m nearest points forming the support domain, referred to as the point star, where m is the numerical empirical parameter. In addition, nodes can be selected and corresponding point stars determined based on criteria such as distance, quadrant, or octant [32].

We take a point $\bar{\mathbf{x}}_0 = (x_0, y_0, t_0)$ in the computational domain as an example to demonstrate the numerical approximation process of the space-time generalized finite difference method. Let u_0 and u_k be the function values of the center point $\bar{\mathbf{x}}_0$ and $\bar{\mathbf{x}}_k = (x_k, y_k, t_k),\ k = 1, 2, \cdots, m$, in the point star. The Taylor series expansion of $\bar{\mathbf{x}}_k$ at $\bar{\mathbf{x}}_0$ is denoted by

$$\begin{aligned} u_k = \ & u_0 + h_k\frac{\partial u_0}{\partial x} + l_k\frac{\partial u_0}{\partial y} + m_k\frac{\partial u_0}{\partial t} + \frac{1}{2}h_k^2\frac{\partial^2 u_0}{\partial x^2} + \frac{1}{2}l_k^2\frac{\partial^2 u_0}{\partial y^2} + \frac{1}{2}m_k^2\frac{\partial^2 u_0}{\partial t^2} \\ & + h_k l_k\frac{\partial^2 u_0}{\partial x\partial y} + h_k m_k\frac{\partial^2 u_0}{\partial x\partial t} + l_k m_k\frac{\partial^2 u_0}{\partial y\partial t} + \cdots, \end{aligned}$$

where $h_k = x_k - x_0$, $l_k = y_k - y_0$, $m_k = t_k - t_0$.

By truncating the Taylor series expansion of $\bar{\mathbf{x}}_k$ at $\bar{\mathbf{x}}_0$ after the second derivative, one can define residual function $B(u_0)$ as

$$\begin{aligned} B(u_0) = \sum_{k=1}^{m}\Bigg[w_k\Big(u_0 - u_k + h_k\frac{\partial u_0}{\partial x} + l_k\frac{\partial u_0}{\partial y} + m_k\frac{\partial u_0}{\partial t} + \frac{1}{2}h_k^2\frac{\partial^2 u_0}{\partial x^2} + \frac{1}{2}l_k^2\frac{\partial^2 u_0}{\partial y^2} + \frac{1}{2}m_k^2\frac{\partial^2 u_0}{\partial t^2} \\ + h_k l_k\frac{\partial^2 u_0}{\partial x\partial y} + h_k m_k\frac{\partial^2 u_0}{\partial x\partial t} + l_k m_k\frac{\partial^2 u_0}{\partial y\partial t}\Big)\Bigg]^2, \end{aligned} \tag{2.1}$$

where w_k represents the weighting function at point $\bar{\mathbf{x}}_k$. w_k weights the residuals at point $\bar{\mathbf{x}}_k$ within $B(u_0)$, decreasing as the distance between $\bar{\mathbf{x}}_k$ and $\bar{\mathbf{x}}_0$ increases. There are various choices for the weighting function, such as potential functions, exponential functions, cubic splines, and quartic splines [32]. Benito et al. [32] and Fan and Li [33] proved that different types of weight functions have little impact on the calculation results, and typically choose quartic spline functions to calculate the weights in Eq. (2.1). In this study, we also use the quartic spline function as the weight function w_k.

$$w_k = 1 - 6\Big(\frac{d_k}{d_m}\Big)^2 + 8\Big(\frac{d_k}{d_m}\Big)^3 - 3\Big(\frac{d_k}{d_m}\Big)^4, \tag{2.2}$$

where $d_k = \sqrt{h_k^2 + l_k^2 + m_k^2}$, and d_m is the maximum value of d_k in the point star.

Define

$$D_{u_0} = \left[\frac{\partial u_0}{\partial x}, \frac{\partial u_0}{\partial y}, \frac{\partial u_0}{\partial t}, \frac{\partial^2 u_0}{\partial x^2}, \frac{\partial^2 u_0}{\partial y^2}, \frac{\partial^2 u_0}{\partial t^2}, \frac{\partial^2 u_0}{\partial x \partial y}, \frac{\partial^2 u_0}{\partial x \partial t}, \frac{\partial^2 u_0}{\partial y \partial t}\right]^T_{9\times 1}. \tag{2.3}$$

To minimize the residual $B(u_0)$, we take the partial derivatives of the variance $B(u_0)$ with respect to the unknown variables in the vector D_{u_0}. Then, setting $\frac{\partial B(u_0)}{\partial D_{u_0}} = 0$, we get a system of linear equations $AD_{u_0} = b$, $A = P^T W^2 P$. Here,

$$\begin{aligned}
&P = (P_1, P_2, \cdots, P_m)^T,\\
&P_k = \left(h_k, l_k, m_k, \frac{h_k^2}{2}, \frac{l_k^2}{2}, \frac{m_k^2}{2}, h_k l_k, h_k m_k, l_k m_k\right)^T, \quad k = 1, 2, \cdots, m,\\
&W = diag(w_1, w_2, \cdots, w_m),\\
&b = \left(-\sum_{k=1}^{m} w_k^2 P_k, w_1^2 P_1, \cdots, w_m^2 P_m\right)(u_0, u_1, \cdots, u_m)^T = BU.
\end{aligned} \tag{2.4}$$

Thus, D_{u_0} can be expressed as

$$D_{u_0} = A^{-1} b = A^{-1} BU = EU, \tag{2.5}$$

where $E = A^{-1}B = (e_{i,j})_{9\times(m+1)}$.

Thus, the derivative of u at $\bar{\mathbf{x}}_0$ can be represented by the linear combination of the function values of m adjacent points in the point star, such as

$$\begin{aligned}
&\frac{\partial u_0}{\partial x} := e_{1,1}u_0 + \sum_{k=1}^{m} e_{1,k+1}u_k, \quad \frac{\partial u_0}{\partial y} := e_{2,1}u_0 + \sum_{k=1}^{m} e_{2,k+1}u_k,\\
&\frac{\partial u_0}{\partial t} := e_{3,1}u_0 + \sum_{k=1}^{m} e_{3,k+1}u_k, \quad \frac{\partial^2 u_0}{\partial y^2} := e_{5,1}u_0 + \sum_{k=1}^{m} e_{5,k+1}u_k,
\end{aligned} \tag{2.6}$$

where $e_{i,j}$ is an element of E generated in the formula (2.5) for the center point $\bar{\mathbf{x}}_0$.

By applying the above numerical procedure at each node, the weighted coefficient matrix E is obtained for all nodes in the computational domain. Since the required data are only related to the node $\bar{\mathbf{x}}_0$ and the nodes $\bar{\mathbf{x}}_k = (x_k, y_k, t_k)$, $k = 1, 2, \cdots, m$, in the point star, and the sparse matrices are formed by the space-time generalized finite difference method discretization problem. Additionally, the space-time generalized finite difference method can be easily extended to solve 1D (3D) problems by using Taylor series expansions of functions with two (four) variables.

3. The discrete problem

In this section, we discretize problem (1.1), $\Omega \subset R^2$, and the numerical scheme of problem (1.1) is derived.

Due to the adoption of the space-time generalized finite difference method, the time axis in problem (1.1) is treated as the 3rd spatial axis, with the initial boundary value conditions being

considered as the boundary conditions of the new problem. At $t = T$, $u(\mathbf{x}, T)$ is enforced to satisfy the governing equation, without any imposed boundary conditions or final conditions.

Taking positive integers M_i, M_b, and N, where M_i and M_b represent the number of nodes selected on the interior $\overline{\Omega}/\partial\Omega$ and boundary $\partial\Omega$, respectively, and N represents the number of time partitions. Let $M = M_i + M_b$, $\Omega_s = \{\mathbf{x}_m | \mathbf{x}_m \in \overline{\Omega}, 1 \le m \le M\}$, $\mathring{\Omega}_s = \{\mathbf{x}_m | \mathbf{x}_m \in \Omega, 1 \le m \le M_i\}$, and $\Omega_t = \{t_n | t_n = n\tau, 0 \le n \le N\}$, where $\tau = \frac{T}{N}$ is the size of the temporal steps, and we obtain the point set $\Omega_s \times \Omega_t = \{(\mathbf{x}_m, t_n) | \mathbf{x}_m \in \overline{\Omega}, t_n = n\tau,\ 1 \le m \le M,\ 0 \le n \le N\}$.

The time fractional derivative at point $(\mathbf{x}_m, t_n)$, $1 \le m \le M,\ 1 \le n \le N$, is discretized.

$$
\begin{aligned}
&{}_0^C D_t^\alpha(\mathbf{x}_m, t_n) \\
&= \frac{1}{\Gamma(1-\alpha)} \int_{t_0}^{t_n} \frac{u_t(\mathbf{x}_m, s)}{(t_n - s)^\alpha} ds \\
&= \frac{1}{\Gamma(1-\alpha)} \sum_{k=1}^{n} \int_{t_{k-1}}^{t_k} \frac{u_t(\mathbf{x}_m, s)}{(t_n - s)^\alpha} ds \\
&\approx \frac{1}{\Gamma(1-\alpha)} \sum_{k=1}^{n} \int_{t_{k-1}}^{t_k} \left[\frac{t - t_k}{t_{k-1} - t_k} u_t(\mathbf{x}_m, t_{k-1}) + \frac{t - t_{k-1}}{t_k - t_{k-1}} u_t(\mathbf{x}_m, t_k) \right] (t_n - s)^{-\alpha} ds \\
&= \frac{1}{\Gamma(2-\alpha)} (t_n - t_0)^{1-\alpha} u_t(\mathbf{x}_m, t_0) + \frac{1}{\Gamma(3-\alpha)} \left[a_{n,1} u_t(\mathbf{x}_m, t_0) + \sum_{k=1}^{n-1} (a_{n,k+1} - a_{n,k}) u_t(\mathbf{x}_m, t_k) - a_{n,n} u_t(\mathbf{x}_m, t_n) \right] \\
&:= D_N^\alpha(\mathbf{x}_m, t_n),
\end{aligned}
\tag{3.1}
$$

where

$$
a_{n,k} = \frac{(t_n - t_k)^{2-\alpha} - (t_n - t_{k-1})^{2-\alpha}}{t_k - t_{k-1}}, \quad k = 1, 2, \cdots, n, \quad n = 1, 2, \cdots, N.
$$

Thus, the discrete scheme of Eq (1.1) at the point $(\mathbf{x}_m, t_n)$ is

$$
\begin{aligned}
&\frac{\partial u}{\partial t}(\mathbf{x}_m, t_n) + D_N^\alpha(\mathbf{x}_m, t_n) = \triangle u(\mathbf{x}_m, t_n) + f(\mathbf{x}_m, t_n), \quad (\mathbf{x}_m, t_n) \in \mathring{\Omega}_s \times \Omega_t / \{t_0\}, \\
&a_i u(\mathbf{x}_m, t_n) + b_i \bigtriangledown u(\mathbf{x}_m, t_n) \cdot \mathbf{n} = \phi_i(\mathbf{x}_m, t_n), \quad (\mathbf{x}_m, t_n) \in \Omega_s / \mathring{\Omega}_s \times \Omega_t, \\
&u(\mathbf{x}_m, 0) = u_0(\mathbf{x}_m), \quad \mathbf{x}_m \in \Omega_s.
\end{aligned}
\tag{3.2}
$$

Based on the introduction of the space-time generalized finite difference method in Section 2, the first- or second-order derivatives of u at the point $(\mathbf{x}_m, t_k)$ in Eq (3.2) with respect to space and time are approximated by the function values of the nodes in its point star. The specific details are as follows.

The point set $\Omega_s \times \Omega_t$ is arranged in time layers from $t = 0$ to $t = T$, and the spatial node distribution is the same in different time layers. Let $q = (n-1) \times M + m$, $X_q = (\mathbf{x}_m, t_n)$, and we obtain

$$
\frac{\partial u}{\partial t}(\mathbf{x}_m, t_n) := e_{3,1}^q u_q + \sum_{k=1}^{m} e_{3,k+1}^q u_{qk}, \tag{3.3}
$$

where $e_{i,j}^q$ is the element in the matrix E obtained in Section 2 with X_q as the center node, and u_{qk} represents the kth node out of m nodes that are closer to the node X_q. $u_t(\mathbf{x}_m, t_k)$ can obtain an approximation similar to $\frac{\partial u}{\partial t}(\mathbf{x}_m, t_n)$ in Eq. (3.3).

In addition, we have

$$\begin{aligned} \triangle u(\mathbf{x}_m, t_n) &= \frac{\partial^2 u}{\partial x^2}(\mathbf{x}_m, t_n) + \frac{\partial^2 u}{\partial y^2}(\mathbf{x}_m, t_n) \\ &:= e^q_{4,1} u_q + \sum_{k=1}^{m} e^q_{4,k+1} u_{qk} + e^q_{5,1} u_q + \sum_{k=1}^{m} e^q_{5,k+1} u_{qk}, \end{aligned} \tag{3.4}$$

and

$$\begin{aligned} \triangledown u(\mathbf{x}_m, t_n) \cdot \mathbf{n} &= \frac{\partial u}{\partial x}(\mathbf{x}_m, t_n)(n_m)_x + \frac{\partial u}{\partial y}(\mathbf{x}_m, t_n)(n_m)_y \\ &:= [e^q_{1,1} u_q + \sum_{k=1}^{m} e^q_{1,k+1} u_{qk}](n_m)_x + [e^q_{2,1} u_q + \sum_{k=1}^{m} e^q_{2,k+1} u_{qk}](n_m)_y, \end{aligned} \tag{3.5}$$

where $(n_m)_x$ and $(n_m)_y$ denote the components of the unit normal vector $\mathbf{n}_m$ on the x-axis and y-axis at point $\mathbf{x}_m$, respectively.

4. Numerical examples

In this section, we discuss the solutions of the time fractional mobile/immobile diffusion equation under two scenarios. First, we assume that the solution is sufficiently smooth on the entire solution domain. Second, we assume that the solution has weak singularities at the initial moment (the solution and its first derivative function with respect to variable t are continuous at $t = 0$; however, the second derivative function with respect to variable t exhibits a singularity at $t = 0$). Numerical examples are provided to verify the stability and accuracy of the algorithm under different spatial dimensions and different boundary conditions. To verify the effectiveness of the algorithm, the L^∞ errors and the root mean square errors are defined as follows:

$$\begin{aligned} L^\infty &= \max_{1\le m\le M_i,\ 0\le k\le N} |u^k_m - U^k_m|, \\ RMSE &= \max_{0\le k\le N} \sqrt{\frac{1}{M_i} \sum_{1\le m\le M_i} (u^k_m - U^k_m)^2}. \end{aligned} \tag{4.1}$$

here, M_i denotes the number of spatial internal nodes in a temporal layer. When nodes are uniformly distributed, the convergence order can be defined as follows.

$$Rate = \frac{\log\left(L(h_2)/L(h_1)\right)}{\log(h_2/h_1)}. \tag{4.2}$$

here, L represents L^∞ or RMSE errors, and h_i $(i = 1, 2)$ represents the unidirectional step size. After perturbing uniformly distributed nodes, the convergence order is also computed according to (4.2). We used MATLAB R2020b to calculate numerical examples on a computer with 2.90 GHz processor and 8 G memory running an Intel 8-core i7-10700 CPU.

In the following numerical simulation, unless stated otherwise, the boundary conditions for problem (1.1) are Dirichlet boundary conditions, with $T = 1$. Nodes in the point star are selected based on distance criteria.

4.1. The solution is sufficiently smooth

In this section, the solution is assumed to be sufficiently smooth on the entire solution domain.

4.1.1. One-dimensional situation

Assume that the exact solution of problem (1.1) is $u(x,t) = t^3 \sin(\pi x)$, $\Omega = (0,1)$.

Set $\alpha = 0.6$, with nodes uniformly distributed and the unidirectional step size is h. The space-time generalized finite difference method employs $2n$th-order ($n = 1,2,3,4$) Taylor series expansions, and the calculation results are shown in Table 1. The data indicates that using a second-order Taylor series expansion, both the L^∞ error and the root mean square error exhibit second-order convergence. Employing 4th-order or higher-order Taylor series expansion yields a more accurate solution. However, due to the linear interpolation approximation used for the first-order derivative of the function in the integral term during the discretization of the time fractional derivative, the convergence order is limited to second order.

Table 1. The errors and convergence orders of the $2n$th-order ($n = 1,2,3,4$) Taylor series expansion in the ST-GFDM with $\alpha = 0.6$.

h	2nd-order ($m = 16$)				4th-order ($m = 30$)			
	L^∞	*Rate*	*RMS E*	*Rate*	L^∞	*Rate*	*RMS E*	*Rate*
1/50	5.90×10^{-4}	-	4.17×10^{-4}	-	1.87×10^{-5}	-	1.34×10^{-5}	-
1/60	4.08×10^{-4}	2.0238	2.89×10^{-4}	2.0197	1.31×10^{-5}	1.9802	9.31×10^{-6}	1.9921
1/70	2.99×10^{-4}	2.0161	2.12×10^{-4}	2.0127	9.62×10^{-6}	1.9778	6.85×10^{-6}	1.9872
1/80	2.29×10^{-4}	2.0096	1.62×10^{-4}	2.0070	7.39×10^{-6}	1.9771	5.26×10^{-6}	1.9849
1/90	1.81×10^{-4}	2.0041	1.28×10^{-4}	2.0021	5.86×10^{-6}	1.9772	4.17×10^{-6}	1.9838
1/100	1.46×10^{-4}	1.9994	1.04×10^{-4}	1.9979	4.75×10^{-6}	1.9776	3.38×10^{-6}	1.9834
h	6th-order ($m = 120$)				8th-order ($m = 200$)			
	L^∞	*Rate*	*RMS E*	*Rate*	L^∞	*Rate*	*RMS E*	*Rate*
1/50	1.87×10^{-5}	-	1.34×10^{-5}	-	1.88×10^{-5}	-	1.34×10^{-5}	-
1/60	1.31×10^{-5}	1.9640	9.32×10^{-6}	1.9739	1.31×10^{-5}	1.9733	9.34×10^{-6}	1.9826
1/70	9.65×10^{-6}	1.9704	6.88×10^{-6}	1.9784	9.66×10^{-6}	1.9751	6.88×10^{-6}	1.9830
1/80	7.41×10^{-6}	1.9739	5.28×10^{-6}	1.9808	7.42×10^{-6}	1.9766	5.28×10^{-6}	1.9834
1/90	5.87×10^{-6}	1.9762	4.18×10^{-6}	1.9822	5.88×10^{-6}	1.9778	4.18×10^{-6}	1.9837
1/100	4.77×10^{-6}	1.9778	3.39×10^{-6}	1.9831	4.77×10^{-6}	1.9790	3.39×10^{-6}	1.9843

Next, we investigate the impact of the number of nodes in the point star on the numerical results. In the space-time domain, nodes are uniformly distributed with a single-direction step size of $h = \frac{1}{40}$, and the total number of nodes is 1681. The space-time generalized finite difference method employs $2n$th-order ($n = 1,2,3,4$) Taylor series expansions, for different values of α, and the variation of L^∞ error as the number of nodes m in the point star is shown in Figure 1. The figure indicates that the

space-time generalized finite difference method uses a 2nd-order Taylor series expansion, and the error increases with the number of nodes in the point star increases. The method employs 4th-order Taylor series expansion, and the error does not exhibit clear positive or negative correlation with the number of nodes in the point star. However, the method uses 6th- or 8th-order Taylor series expansions, and the error slightly decreases as the number of nodes in the point star increases.

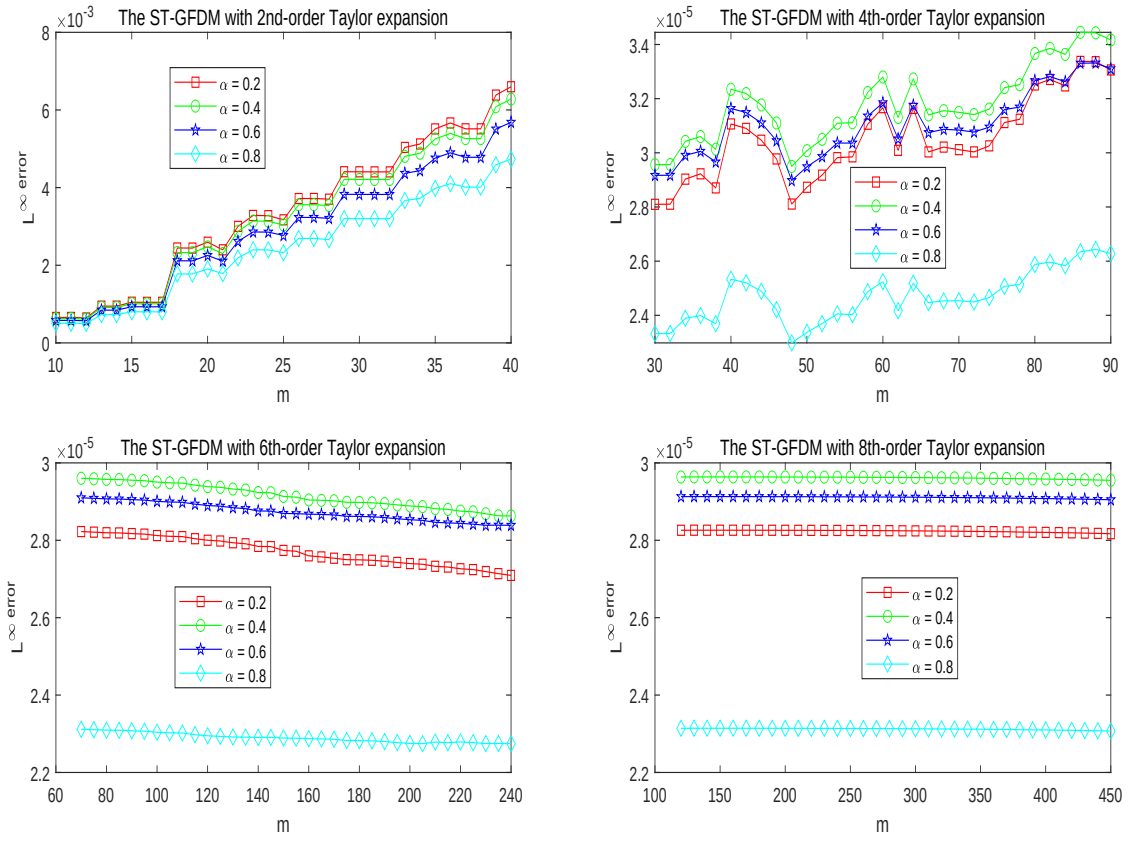

Figure 1. The variation of L^∞ error with the number of nodes m in the point star.

4.1.2. Two-dimensional situation

Assume that the exact solution of problem (1.1) is $u(x, y, t) = t^3 \sin(\pi x) \sin(\pi y)$.

Consider the square domain $\Omega = (0, 1)^2$. We will investigate the impact of spatial nodes on errors under different distributions. First, we consider nodes uniformly distributed, selecting positive integers M and N to represent the number of subdivisions in the spatial and temporal directions, respectively. The spatial step size in one direction is denoted as $h = \frac{1}{M}$, and the temporal step size as $\tau = \frac{1}{N}$. This yields spatially uniform grid $\Omega_h = \{(x_i, y_j)|x_i = ih, y_j = jh, 0 \le i, j \le M\}$ and the set of temporal nodes $\Omega_\tau = \{t_n|t_n = n\tau, 0 \le n \le N\}$.

Then, by employing inline MATLAB code to introduce minor random perturbations for the mesh, we define the spatial interior non-uniform grid points $\widetilde{\Omega}_h = \{(\tilde{x}_i, \tilde{y}_j)|\tilde{x}_i = x_i + h\eta(2\lambda_{i,j} - 1), \tilde{y}_j = y_j + h\eta(2\lambda_{i,j} - 1), 1 \le i, j \le M - 1\}$, where $\lambda_{i,j}$ represents a random number between 0 and 1, and $0 < \eta \le 1$

is the mesh parameter used to adjust the non-uniformity of the mesh. Obviously, when $\eta = 0$, $\widetilde{\Omega}_h$ indicates a uniform grid. As η increases, the non-uniformity of the spatial grid gradually increases and reaches its maximum when $\eta = 1$. It is noteworthy that the non-uniform grid is generated randomly. Under different distributions of internal nodes in space, the nodes at the spatial boundaries and in the time domain remain uniformly distributed. Define $\Omega_h = \widetilde{\Omega}_h \cup \partial\Omega_h$, where $\partial\Omega_h$ denotes the set of spatial boundary nodes. When $M = 20$, $\eta = 0, 0.1, 0.2, 0.3, 0.4, 0.5$, the distribution of spatial nodes at each time layer is shown in Figure 2, where the solid and hollow pentagrams represent the internal and boundary nodes, respectively.

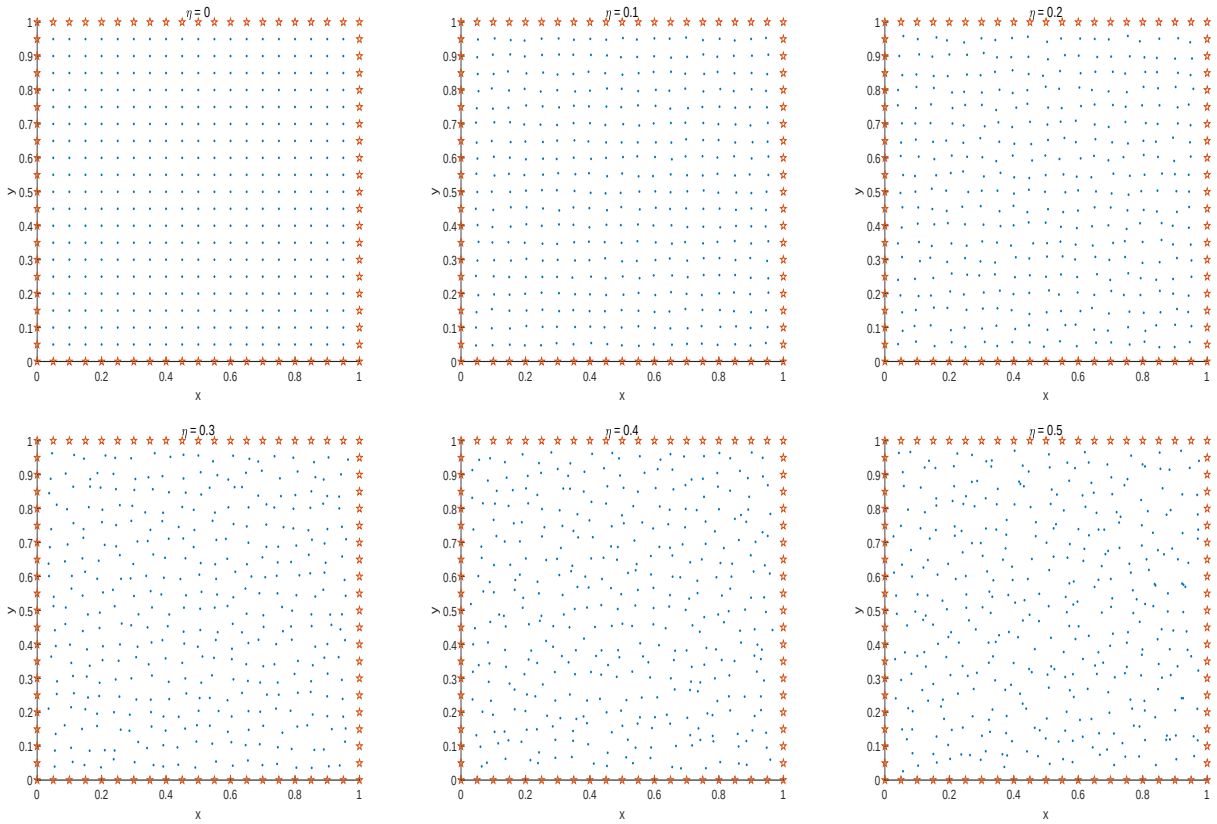

Figure 2. The distribution of nodes with $M = 20$.

In the next calculation, we set $\alpha = 0.6$, and the number of time subdivisions is the same as the number of subdivisions in space single direction. The space-time generalized finite difference method employs second-order Taylor series expansion, the computed L^∞ errors, root mean square errors, and convergence orders under different spatial node distributions are listed in Table 2. The data show that for non-uniform mesh parameters $\eta = 0, 0.1, 0.2, 0.3$, setting the number of nodes $m = 30$ in the point star yields second-order convergence result. For $\eta = 0.4, 0.5$, the increase in non-uniformity of the node distribution requires the linear combination of more node function values in the point star to approximate the derivatives, in order to achieve the desired results. For $\eta = 0.4, 0.5$, the node distribution is more irregular, and it is necessary to increase the number of nodes in the point star to achieve good calculation results. If the value of η increases further, the non-uniformity of the node distribution becomes more pronounced, with some nodes clustering together and some areas having

sparse nodes. When selecting nodes in a point star based on distance, most chosen nodes may become concentrated in a specific direction relative to the central point, potentially leading to inaccuracies in the resulting computations.

Table 2. The errors and convergence orders of second-order Taylor series expansion in the ST-GFDM with different nodes.

h	$\eta = 0,\ m = 30$				$\eta = 0.1,\ m = 30$			
	L^∞	*Rate*	*RMS E*	*Rate*	L^∞	*Rate*	*RMS E*	*Rate*
1/5	8.02×10^{-2}	-	3.85×10^{-2}	-	8.40×10^{-2}	-	3.95×10^{-2}	-
1/10	2.17×10^{-2}	1.8874	1.08×10^{-2}	1.8341	2.18×10^{-2}	1.9484	1.08×10^{-2}	1.8662
1/15	9.33×10^{-3}	2.0788	4.72×10^{-3}	2.0380	9.36×10^{-3}	2.0825	4.74×10^{-3}	2.0389
1/20	5.20×10^{-3}	2.0303	2.60×10^{-3}	2.0668	5.27×10^{-3}	1.9991	2.62×10^{-3}	2.0528
1/25	3.26×10^{-3}	2.0925	1.64×10^{-3}	2.0751	3.27×10^{-3}	2.1389	1.64×10^{-3}	2.1141
1/30	2.24×10^{-3}	2.0535	1.12×10^{-3}	2.0758	2.26×10^{-3}	2.0330	1.12×10^{-3}	2.0629
h	$\eta = 0.2,\ m = 30$				$\eta = 0.3,\ m = 30$			
	L^∞	*Rate*	*RMS E*	*Rate*	L^∞	*Rate*	*RMS E*	*Rate*
1/5	7.78×10^{-2}	-	3.89×10^{-2}	-	9.21×10^{-2}	-	4.43×10^{-2}	-
1/10	2.23×10^{-2}	1.8056	1.13×10^{-2}	1.7861	1.95×10^{-2}	2.2396	9.95×10^{-3}	2.1547
1/15	9.45×10^{-3}	2.1125	4.78×10^{-3}	2.1219	8.84×10^{-3}	1.9514	4.58×10^{-3}	1.9140
1/20	5.12×10^{-3}	2.1310	2.59×10^{-3}	2.1269	5.15×10^{-3}	1.8743	2.57×10^{-3}	2.0092
1/25	3.24×10^{-3}	2.0534	1.62×10^{-3}	2.1144	3.34×10^{-3}	1.9407	1.69×10^{-3}	1.8659
1/30	2.28×10^{-3}	1.9109	1.17×10^{-3}	1.7870	2.29×10^{-3}	2.0691	1.16×10^{-3}	2.0791
h	$\eta = 0.4,\ m = 50$				$\eta = 0.5,\ m = 80$			
	L^∞	*Rate*	*RMS E*	*Rate*	L^∞	*Rate*	*RMS E*	*Rate*
1/5	1.21×10^{-1}	-	5.79×10^{-2}	-	1.70×10^{-1}	-	8.52×10^{-2}	-
1/10	3.26×10^{-2}	1.8910	1.65×10^{-2}	1.8093	4.99×10^{-2}	1.7678	2.38×10^{-2}	1.8393
1/15	1.42×10^{-2}	2.0472	7.06×10^{-3}	2.0971	2.29×10^{-2}	1.9164	1.12×10^{-2}	1.8546
1/20	7.90×10^{-3}	2.0459	3.95×10^{-3}	2.0166	1.33×10^{-2}	1.8854	6.58×10^{-3}	1.8544
1/25	5.15×10^{-3}	1.9163	2.56×10^{-3}	1.9522	9.05×10^{-3}	1.7373	4.37×10^{-3}	1.8338
1/30	3.55×10^{-3}	2.0462	1.76×10^{-3}	2.0338	6.45×10^{-3}	1.8561	3.19×10^{-3}	1.7264

Consider employing the eight quadrant criterion for node selection in a point star. Specifically, when $\eta = 0$, four nodes are selected from each quadrant, yielding a total of 32 nodes per point star; when $\eta = 0.5$, five nodes are selected per quadrant, resulting in 40 nodes per point star. The computed L^∞ errors, root mean square errors, and convergence orders are presented in Table 3. The results indicate that for uniformly distributed nodes, the eight-quadrant criterion is less accurate than selecting

nodes based on distance, as it ignores nearby nodes on the coordinate plane. Conversely, for unevenly distributed nodes, the eight-quadrant criterion uniformly gathers nodes from all directions, and when the number of nodes in the point star is not large, a more accurate solution can be obtained.

Table 3. The errors and convergence orders of second-order Taylor series expansion in the ST-GFDM with different nodes.

h	$\eta = 0,\ m = 32$				$\eta = 0.5,\ m = 40$			
	L^∞	*Rate*	*RMS E*	*Rate*	L^∞	*Rate*	*RMS E*	*Rate*
1/5	1.37×10^{-1}	-	6.54×10^{-2}	-	1.44×10^{-1}	-	7.86×10^{-2}	-
1/10	3.92×10^{-2}	1.8035	2.02×10^{-2}	1.6938	4.08×10^{-2}	1.8213	2.10×10^{-2}	1.9043
1/15	1.89×10^{-2}	1.7954	9.53×10^{-3}	1.8521	1.99×10^{-2}	1.7672	1.01×10^{-2}	1.8165
1/20	1.21×10^{-2}	1.5544	6.07×10^{-3}	1.5667	1.26×10^{-2}	1.6017	6.12×10^{-3}	1.7274
1/25	8.63×10^{-3}	1.5135	4.31×10^{-3}	1.5329	8.70×10^{-3}	1.6452	4.38×10^{-3}	1.4993
1/30	6.33×10^{-3}	1.6942	3.19×10^{-3}	1.6564	6.33×10^{-3}	1.7459	3.17×10^{-3}	1.7735

Consider problem (1.1) with Dirichlet and Neumann mixed boundary conditions

$$u(\mathbf{x},t) = \phi_1(\mathbf{x},t), \quad \mathbf{x}\in\partial\Omega_1, \quad t\in(0,1],$$
$$\bigtriangledown u(\mathbf{x},t)\cdot\mathbf{n} = \phi_2(\mathbf{x},t), \quad \mathbf{x}\in\partial\Omega_2, \quad t\in(0,1],$$

where $\partial\Omega_2 = \{(x,y)|x = 1, y\in(0,1)\}$, $\partial\Omega_1 = \partial\Omega/\partial\Omega_2$. The space-time generalized finite difference method adopts second-order Taylor series expansion, and the number of nodes in the point star is $m = 30$. The computed root mean square errors and convergence orders are listed in Table 4. The data shows that the algorithm can effectively calculate mixed boundary condition problems.

Table 4. The errors and convergence orders of second-order Taylor series expansion in the ST-GFDM with mixed boundary conditions.

h	$\alpha = 0.2$		$\alpha = 0.4$		$\alpha = 0.6$		$\alpha = 0.8$	
	RMS E	*Rate*	*RMS E*	*Rate*	*RMS E*	*Rate*	*RMS E*	*Rate*
1/5	5.81×10^{-2}	-	5.71×10^{-2}	-	5.58×10^{-2}	-	5.45×10^{-2}	-
1/10	1.74×10^{-2}	1.7410	1.69×10^{-2}	1.7540	1.63×10^{-2}	1.7776	1.55×10^{-2}	1.8141
1/15	7.61×10^{-3}	2.0381	7.39×10^{-3}	2.0420	7.08×10^{-3}	2.0551	6.67×10^{-3}	2.0806
1/20	4.20×10^{-3}	2.0658	4.08×10^{-3}	2.0644	3.90×10^{-3}	2.0693	3.66×10^{-3}	2.0851
1/25	2.65×10^{-3}	2.0706	2.57×10^{-3}	2.0666	2.46×10^{-3}	2.0666	2.30×10^{-3}	2.0761
1/30	1.81×10^{-3}	2.0683	1.77×10^{-3}	2.0631	1.69×10^{-3}	2.0602	1.58×10^{-3}	2.0656

Consider a circular domain $\Omega = \{(x,y)|(x-0.5)^2 + (y-0.5)^2 < 0.5^2\}$, with uniformly distributed spatiotemporal nodes. The unidirectional step size is h, and adjacent node arc lengths $s = h$ on the circumference. We set $h = \frac{1}{10}$, and the distribution of spatiotemporal nodes observed from different

angles is shown in Figure 3. Here, blue dots represent nodes on $\overline{\Omega} \times \{t = 1\}$, green dots represent nodes on $\overline{\Omega} \times \{t = 0\}$ and $\partial\Omega \times (0, T)$, and black dots represent nodes in $\Omega \times (0, T)$.

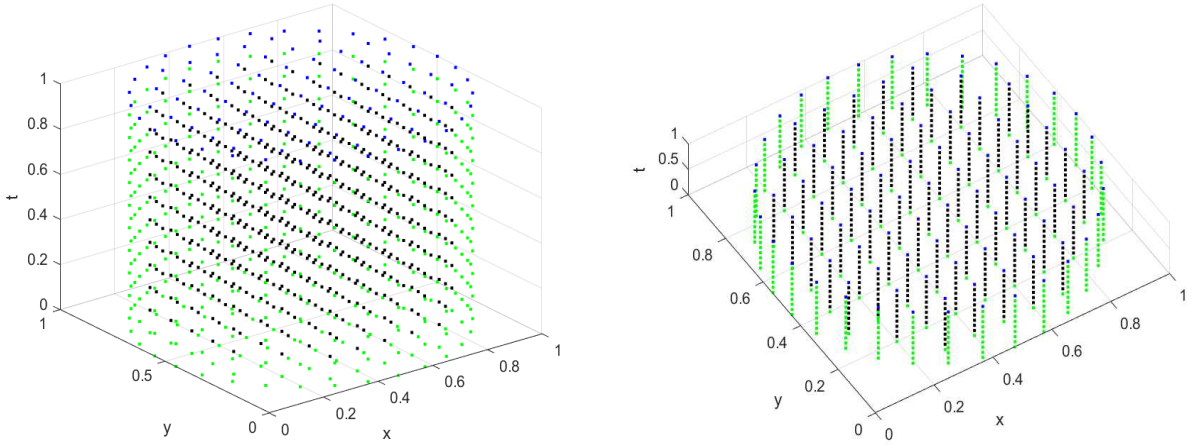

Figure 3. The distribution of nodes in the space-time domain.

We set $\alpha = 0.6$, the generalized finite difference method uses second-order Taylor series expansion, and the number of nodes in the point star domain is $m = 30$. The computed errors, convergence orders, required time (in seconds), and the speedup ratios are listed in Table 5, where the values in the "Serial" and "Parallel" columns represent the required time for serial and parallel computing, respectively. The parallel algorithm is computed using the MATLAB Parallel Computing Toolbox, which is connected to the parallel pool (number of workers: 8). The data indicates that the generalized finite difference method can also handle problems in circular domains. When the number of nodes is small, the advantages of multi-core processing are not fully demonstrated. However, as the number of nodes increases, the time required for parallel algorithm computation is significantly less than that of the serial algorithm.

Table 5. The errors, convergence orders, computation time, and speedup ratios of second-order Taylor series expansion in the ST-GFDM.

h	L^∞	*Rate*	*RMSE*	*Rate*	Serial	Parallel	Ratios
1/5	7.16×10^{-2}	-	4.48×10^{-2}	-	0.1779	0.2447	0.7270
1/10	2.07×10^{-2}	1.7894	1.16×10^{-2}	1.9551	1.7960	0.8526	2.1065
1/15	8.92×10^{-3}	2.0775	4.86×10^{-3}	2.1343	21.1992	3.8431	5.5162
1/20	5.10×10^{-3}	1.9408	2.77×10^{-3}	1.9534	142.3594	22.5884	6.3023
1/25	3.20×10^{-3}	2.0944	1.74×10^{-3}	2.0843	598.4483	133.8610	4.4707
1/30	2.17×10^{-3}	2.1204	1.18×10^{-3}	2.1431	1776.0657	465.6100	3.8145

Consider an amoeba-like irregular domain whose boundary is determined by the function $\rho(\theta)$, with $\rho(\theta) = \frac{1}{2}[\exp(\sin\theta)\sin^2(2\theta) + \exp(\cos\theta)\cos^2(2\theta)]$, $0 \le \theta \le 2\pi$, and then $x(\theta) = \rho(\theta)\cos(\theta)$ and $y(\theta) = \rho(\theta)\sin(\theta)$. The nodes in the space-time domain are uniformly distributed, with a unidirectional

step size of h. At the same time layer, the Euclidean distance between two adjacent nodes on the boundary is $\sqrt{2}h$. When $h = \frac{1}{10}$, the spatial node distribution at each time layer and spatiotemporal domain node distribution are shown in Figure 4. The space-time generalized finite difference method adopts second-order Taylor series expansion, with a node number of $m = 30$ in the point star. The computed root mean square errors and convergence orders for different α are shown in Table 6. The data shows that the proposed space-time generalized finite difference method can also effectively solve irregular domain problems.

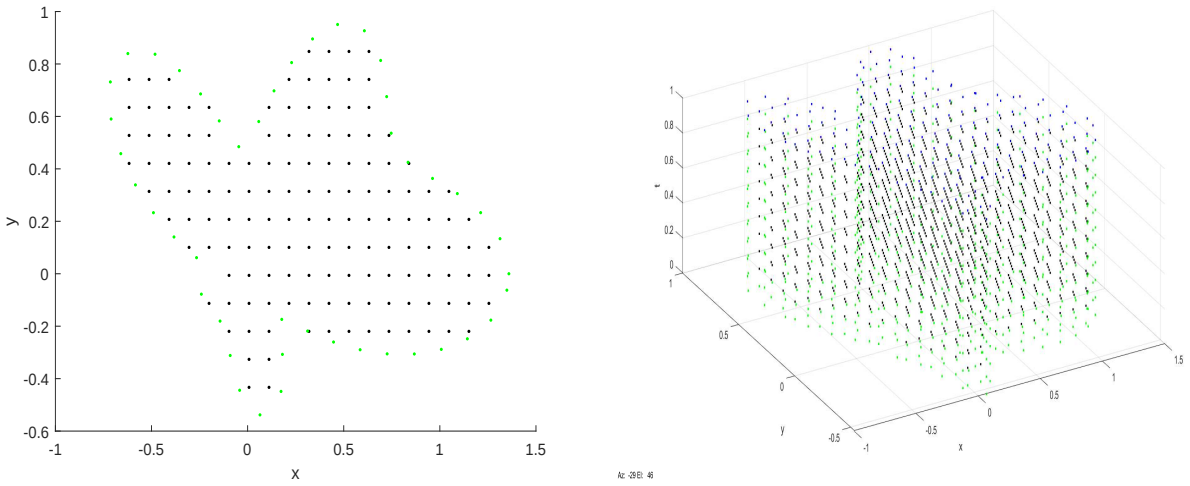

Figure 4. The distribution of nodes in an amoeba-like irregular domain.

Table 6. The errors and convergence orders of second-order Taylor series expansion in the ST-GFDM on an amoeba-like irregular domain.

h	$\alpha = 0.2$		$\alpha = 0.4$		$\alpha = 0.6$		$\alpha = 0.8$	
	RMS E	*Rate*	*RMS E*	*Rate*	*RMS E*	*Rate*	*RMS E*	*Rate*
1/5	3.14×10^{-2}	-	3.13×10^{-2}	-	3.12×10^{-2}	-	3.12×10^{-2}	-
1/10	8.31×10^{-3}	1.9196	8.13×10^{-3}	1.9430	7.89×10^{-3}	1.9828	7.56×10^{-3}	2.0446
1/15	3.48×10^{-3}	2.1497	3.37×10^{-3}	2.1733	3.22×10^{-3}	2.2124	3.02×10^{-3}	2.2661
1/20	2.11×10^{-3}	1.7277	2.05×10^{-3}	1.7210	1.95×10^{-3}	1.7313	1.80×10^{-3}	1.7870
1/25	1.37×10^{-3}	1.9364	1.33×10^{-3}	1.9466	1.26×10^{-3}	1.9664	1.15×10^{-3}	2.0075
1/30	9.43×10^{-4}	2.0589	9.09×10^{-4}	2.0896	8.54×10^{-4}	2.1360	7.71×10^{-4}	2.2023

4.2. The solution with weak singularity

In this subsection, assuming that the solution of problem (1.1) has weak singularity at the initial moment satisfying $\frac{\partial^k u}{\partial t^k} \in O(t^{1+\alpha-k})$, $k = 0, 1, 2, 3$, when the center point is $(\mathbf{x}_i, t_0)$, $i = 1, 2, \cdots, M$, the points in the point star cannot be approximated by higher-order Taylor expansion. Specifically, the values of the derivatives at point $(\mathbf{x}_i, t_0)$ cannot be approximated using weighted values of the function

in the point star. In this case, it is necessary to make appropriate adjustments to the aforementioned space-time generalized finite difference method, defining

$$\frac{\partial u}{\partial t}(\mathbf{x}_i, t_0) = \frac{\partial u}{\partial t}(\mathbf{x}_i, t_1) + O(h). \tag{4.3}$$

Additionally, for the point $(\mathbf{x}_i, t_k)$, $i = 1, 2, \cdots, M$, $k \neq 0$, if its point star contains the point $(\mathbf{x}_j, t_0)$, $j = 1, 2, \cdots, M$, then $u(\mathbf{x}_j, t_0) = u(\mathbf{x}_i, t_k) + (x_j - x_i)\frac{\partial u}{\partial x}(\mathbf{x}_i, t_k) + (y_j - y_i)\frac{\partial u}{\partial y}(\mathbf{x}_i, t_k) + (t_0 - t_k)\frac{\partial u}{\partial t}(\mathbf{x}_i, t_k) + O(h^2)$, for $d = 2$.

4.2.1. One-dimensional situation

Assume that the exact solution of problem (1.1) is $u(x, t) = \left[1 + \frac{t^{1+\alpha}}{\Gamma(2+\alpha)}\right] \sin(\pi x)$, $\Omega = (0, 1)$.

The space-time generalized finite difference method employs second-order Taylor series expansion with uniformly distributed nodes, and the number of nodes $m = 30$ in the point star. The computed L^∞ errors, root mean square errors, and convergence orders for $\alpha = 0.2, 0.4, 0.6, 0.8$ are listed in Table 7. The data indicates that when the solution has weak singularity at the initial moment, for different values of α, as the number of nodes increases, the error gradually decreases, and the convergence order tends toward 2.

Table 7. The errors and convergence rates of second-order Taylor series expansion in the ST-GFDM with $m = 30$.

h	$\alpha = 0.2$				$\alpha = 0.4$			
	L^∞	*Rate*	*RMSE*	*Rate*	L^∞	*Rate*	*RMSE*	*Rate*
1/50	2.34×10^{-3}	-	1.62×10^{-3}	-	2.24×10^{-3}	-	1.55×10^{-3}	-
1/60	1.64×10^{-3}	1.9247	1.15×10^{-3}	1.9029	1.58×10^{-3}	1.9134	1.10×10^{-3}	1.8923
1/70	1.22×10^{-3}	1.9323	8.53×10^{-4}	1.9138	1.17×10^{-3}	1.9213	8.20×10^{-4}	1.9034
1/80	9.43×10^{-4}	1.9369	6.60×10^{-4}	1.9211	9.08×10^{-4}	1.9264	6.36×10^{-4}	1.9113
1/90	7.50×10^{-4}	1.9399	5.26×10^{-4}	1.9262	7.23×10^{-4}	1.9301	5.07×10^{-4}	1.9170
1/100	6.11×10^{-4}	1.9418	4.29×10^{-4}	1.9298	5.90×10^{-4}	1.9328	4.14×10^{-4}	1.9214
h	$\alpha = 0.6$				$\alpha = 0.8$			
	L^∞	*Rate*	*RMSE*	*Rate*	L^∞	*Rate*	*RMSE*	*Rate*
1/50	2.04×10^{-3}	-	1.42×10^{-3}	-	1.77×10^{-3}	-	1.23×10^{-3}	-
1/60	1.44×10^{-3}	1.9097	1.01×10^{-3}	1.8889	1.25×10^{-3}	1.9230	8.71×10^{-4}	1.9027
1/70	1.07×10^{-3}	1.9165	7.50×10^{-4}	1.8990	9.28×10^{-4}	1.9280	6.49×10^{-4}	1.9109
1/80	8.31×10^{-4}	1.9211	5.82×10^{-4}	1.9062	7.17×10^{-4}	1.9311	5.02×10^{-4}	1.9166
1/90	6.62×10^{-4}	1.9243	4.65×10^{-4}	1.9116	5.71×10^{-4}	1.9333	4.01×10^{-4}	1.9208
1/100	5.40×10^{-4}	1.9269	3.80×10^{-4}	1.9158	4.66×10^{-4}	1.9348	3.27×10^{-4}	1.9240

To visually observe the variation of error with the number of nodes, we set the nodes uniformly distributed and the number of subdivisions in a single direction as M. The space-time generalized finite difference method employs second-order Taylor series expansion, with the number of nodes $m = 30$ in the point star. The results of the maximum errors and root mean square errors under different subdivisions are depicted in Figure 5 for tests $\alpha = 0.1, 0.3, 0.5, 0.7, 0.9$. Figure 5 shows that for different α, the error decreases as the total number of nodes increases, and the convergence order tends toward 2.

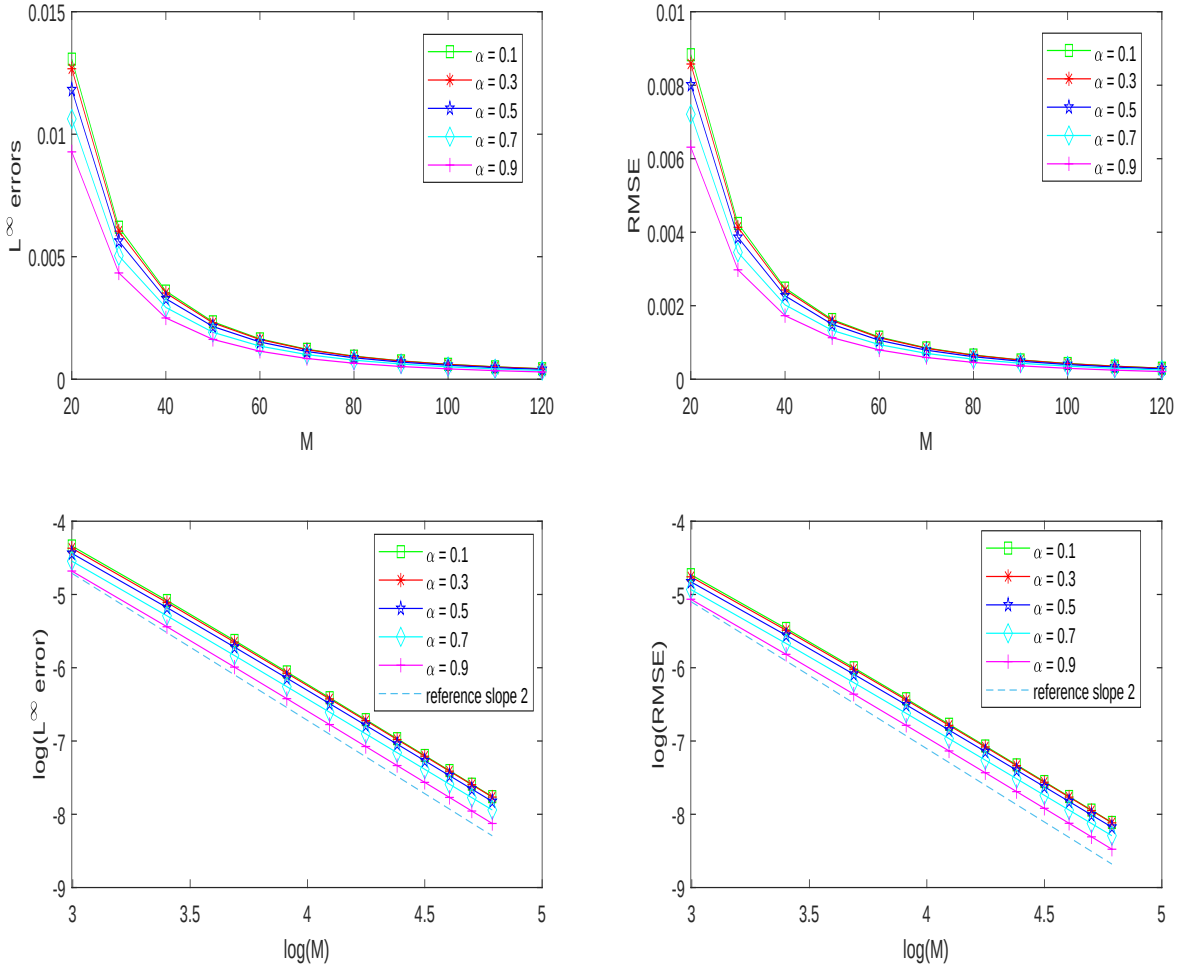

Figure 5. The errors on log-log scale of second-order Taylor series expansion in the ST-GFDM with $m = 30$.

We set $\alpha = 0.6$, with uniformly distributed nodes in the space-time domain and a unidirectional step size of $h = \frac{1}{60}$. The space-time generalized finite difference method uses second-order Taylor series expansion, with the number of nodes $m = 30$ in the point star. The numerical and exact solutions at different times are shown in Figure 6. The image shows that the numerical solution fits the exact solution very well.

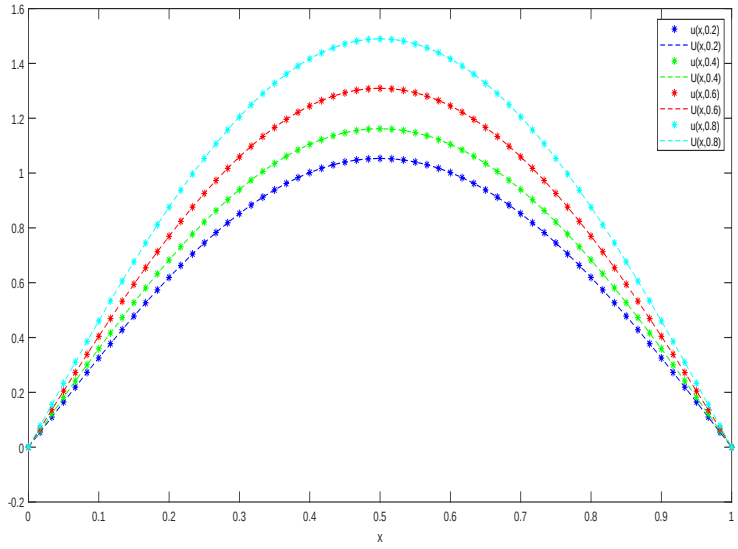

Figure 6. Numerical and exact solutions at different times.

We set $\alpha = 0.6$, the space-time generalized finite difference method adopts second-order Taylor series expansion, the number of nodes in the point star is $m = 30$, the nodes in the space-time domain are uniformly distributed, and the number of partitions in each direction is $M = 20, 40, 60, 80, 100, 120$. Select the quartic spline function (2.2), potential function: $w_k = \frac{1}{d_k^2}$, exponential function: $w_k = e^{-10d_k^2}$, and cubic spline function:

$$w_k = \begin{cases} \left(\frac{2}{3} - 4d_k^2 + 4d_k^3\right)^2, & 0 \le d_k \le 0.25, \\ \left(\frac{4}{3} - 4d_k + 4d_k^3 - \frac{4}{3}d_k^3\right)^2, & 0.25 < d_k \le 0.5, \\ e^{-10}, & d_k > 0.5, \end{cases}$$

as the weight function w_k in formula (2.1). The computed errors are shown in Figure 7. We can observe that the accuracy of the computed results varies depending on the choice of weight function, and the quartic spline function as the weight function yields more accurate results.

4.2.2. Two-dimensional situation

Assume that the exact solution of problem (1.1) is $u(x, y, t) = \left[1 + \frac{t^{1+\alpha}}{\Gamma(2+\alpha)}\right] \sin(\pi x) \sin(\pi y)$, $\Omega = (0, 1)^2$.

The nodes are uniformly distributed with a total number of 9261, meaning that the step length in a single direction is $h = \frac{1}{20}$. The space-time generalized finite difference method employs second-order Taylor series expansion, and the number of nodes $m = 30$ in the point star domain. Setting $\alpha = 0.6$, the exact solution, numerical solution, and absolute errors at various spatial points at $T = 1$ are shown in Figure 8. The figure shows that the trends of the exact and numerical solutions are similar, with absolute errors on the order of 10^{-3}, indicating that the numerical solution can approximate the exact solution very well.

The space-time generalized finite difference method employs second-order Taylor series expansion with uniformly distributed nodes. In the point star, the number of nodes is set to $m = 30$. The computed

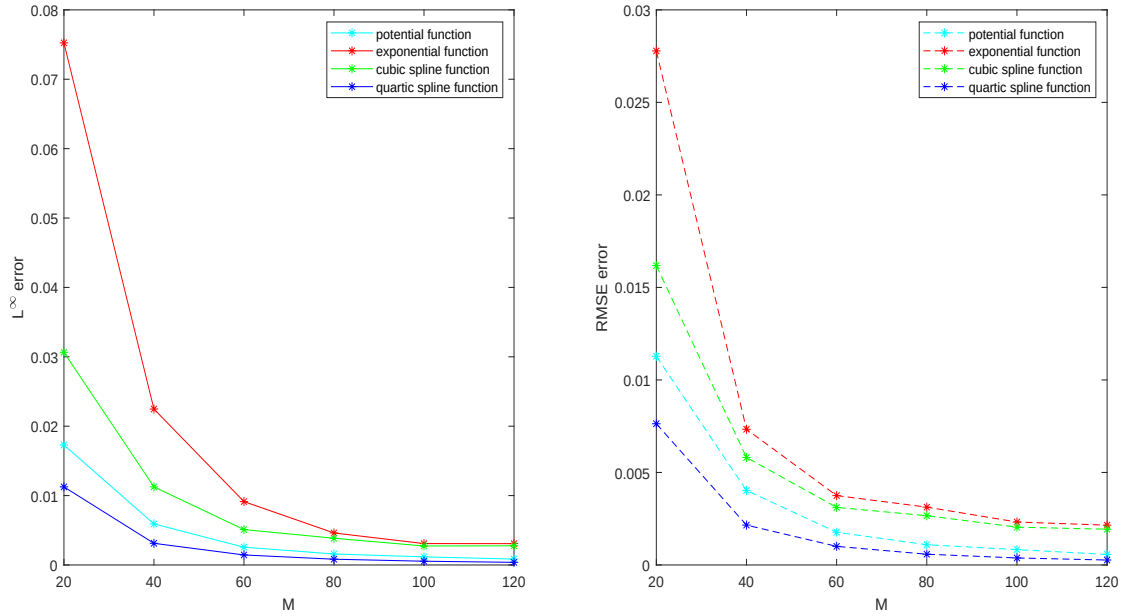

Figure 7. The computed errors adopting different weight functions in ST-GFDM.

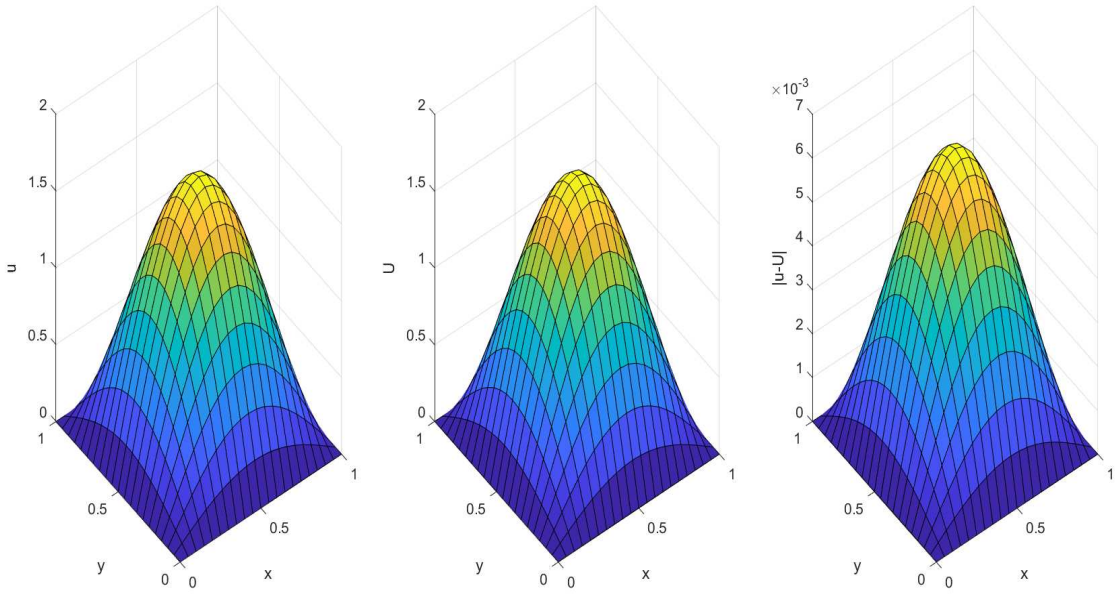

Figure 8. The exact solution, numerical solution, and absolute error with $\alpha = 0.6$ at $T = 1$.

L^∞ errors, root mean square errors, and convergence orders for different α are shown in Table 8. The data indicates that second-order convergence can be achieved for various values of α.

Table 8. The errors and convergence rates of second-order Taylor series expansion in the ST-GFDM with $m = 30$.

h	$\alpha = 0.2$				$\alpha = 0.4$			
	L^∞	*Rate*	*RMS E*	*Rate*	L^∞	*Rate*	*RMS E*	*Rate*
1/5	1.16×10^{-1}	-	5.56×10^{-2}	-	1.10×10^{-1}	-	5.28×10^{-2}	-
1/10	3.10×10^{-2}	1.9055	1.54×10^{-2}	1.8519	2.93×10^{-2}	1.9089	1.46×10^{-2}	1.8555
1/15	1.34×10^{-2}	2.0739	6.75×10^{-3}	2.0331	1.27×10^{-2}	2.0745	6.40×10^{-3}	2.0337
1/20	7.48×10^{-3}	2.0127	3.74×10^{-3}	2.0492	7.10×10^{-3}	2.0116	3.55×10^{-3}	2.0481
1/25	4.72×10^{-3}	2.0681	2.37×10^{-3}	2.0506	4.48×10^{-3}	2.0664	2.25×10^{-3}	2.0490
1/30	3.26×10^{-3}	2.0255	1.63×10^{-3}	2.0478	3.09×10^{-3}	2.0239	1.55×10^{-3}	2.0462
h	$\alpha = 0.6$				$\alpha = 0.8$			
	L^∞	*Rate*	*RMS E*	*Rate*	L^∞	*Rate*	*RMS E*	*Rate*
1/5	1.04×10^{-1}	-	4.97×10^{-2}	-	9.68×10^{-2}	-	4.64×10^{-2}	-
1/10	2.74×10^{-2}	1.9210	1.36×10^{-2}	1.8678	2.52×10^{-2}	1.9422	1.25×10^{-2}	1.8893
1/15	1.18×10^{-2}	2.0818	5.95×10^{-3}	2.0410	1.08×10^{-2}	2.0961	5.45×10^{-3}	2.0552
1/20	6.59×10^{-3}	2.0148	3.30×10^{-3}	2.0513	6.02×10^{-3}	2.0227	3.01×10^{-3}	2.0591
1/25	4.16×10^{-3}	2.0671	2.09×10^{-3}	2.0497	3.79×10^{-3}	2.0707	1.91×10^{-3}	2.0532
1/30	2.87×10^{-3}	2.0234	1.44×10^{-3}	2.0457	2.62×10^{-3}	2.0245	1.31×10^{-3}	2.0468

To visually observe the variation of errors with time, we set nodes uniformly distributed with a single directional step size of $\frac{1}{15}$. The space-time generalized finite difference method employs second-order Taylor series expansion, with the number of nodes $m = 30$ in the point star. For $\alpha = 0.2, 0.4, 0.6, 0.8$, the relative error's variation at different moments from $t = 0$ to $t = 2$ is shown in Figure 9, where the relative errors at the moment t_n are defined as

$$relative\ errors = \sqrt{\frac{\sum\limits_{1\le m\le M_i} |u_m^n - U_m^n|^2}{\sum\limits_{1\le m\le M_i} |u_m^n|^2}}.$$

The figure shows that the error is convergent over time.

We set $\alpha = 0.6$, with uniformly distributed nodes in the space-time domain and a single directional step size of h. We adopt the space-time generalized finite difference method proposed in this paper, the averaged L1 generalized finite difference method (AL1-GFDM: the time derivative is discretized using the averaged L1 method, and the spatial derivative is approximated using the generalized finite difference method), and the mean L1 finite difference method (AL1-FDM) [34] to compute problem (1.1), respectively, where both the generalized finite difference method and the space-time

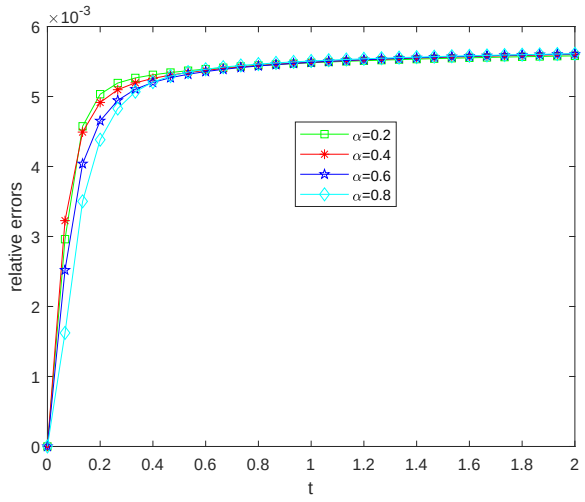

Figure 9. The relative errors of second-order Taylor series expansion in the ST-GFDM with different α.

generalized finite difference method use second-order Taylor series expansion, and the number of nodes in the point star is $m = 30$. The root mean square errors, convergence orders, and required time are listed in Table 9. The data shows that when the discrete points are the same, the space-time generalized finite difference method takes longer to calculate compared to the other two algorithms, which is its disadvantage. The accuracy of the calculation results is between the average L1 finite difference method and the average L1 generalized finite difference method. However, compared to the other two methods, the space-time generalized finite difference method does not require discussion on the choice of time step to ensure the stability of the algorithm. Compared with the averaged L1 finite difference method, it does not require grid generation, making it easier to calculate complex domains and problems with complex boundary conditions.

Table 9. The root mean square errors, convergence rates, and computation time (in seconds) obtained for different numerical algorithms.

h	ST-GFDM		AL1-GFDM		AL1-FDM	
	$RMSE$	$Rate$	$RMSE$	$Rate$	$RMSE$	$Rate$
$1/5$	4.97×10^{-2}	-	1.04×10^{-1}	-	3.06×10^{-2}	-
$1/10$	1.36×10^{-2}	1.8678	2.38×10^{-2}	2.1199	6.71×10^{-3}	2.1919
$1/15$	5.95×10^{-3}	2.0410	1.09×10^{-2}	1.9306	2.86×10^{-3}	2.1005
$1/20$	3.30×10^{-3}	2.0513	6.22×10^{-3}	1.9467	1.58×10^{-3}	2.0707
$Time(s)$	25.5156		8.5781		0.2969	

5. Discussion

In this paper, we numerically investigate the one- and two-dimensional time-fractional mobile/immobile diffusion equation by the space-time generalized finite difference method.

We tested the effect of several parameter variables such as fractional order α, the number of nodes m in the point star, and the total number of nodes, as well as the distribution of nodes, on the calculation results. The data indicates that within a certain range, m has minor impact on the numerical results, while the total number of nodes is negatively correlated with the error. When the irregularity of node distribution is controlled within a certain range, an accurate numerical solution can also be obtained, which has slight effect on the error and convergence order. As for the parameter α, it has slightly different effects on the computational results depending on the control equations.

The space-time generalized finite difference method is based on Taylor series expansion and least squares. We present computational results using $2n$th-order $(n = 1, 2, 3, 4)$ Taylor series expansions, which demonstrate that higher-order expansions yield more accurate numerical results. However, due to the linear interpolation approximation of the first derivative in the integrand of the fractional derivative, the final convergence order is also 2. Additionally, when the solution of the equation is sufficiently smooth, the space-time generalized finite difference method can be directly applied. But when the solution has weak singularities, we need to make appropriate modifications for the algorithm. Regardless of whether the solution is sufficiently smooth, achieving higher convergence orders requires further research.

The space-time generalized finite difference method can effectively solve problems with irregular domains and complex boundary conditions. The calculation time is mainly related to the total number of nodes, the number of nodes in the point star, and the number of Taylor series expansion terms, and does not increase due to the irregularity of the domain or the complexity of the boundary conditions. In this paper, parallel computing is considered to effectively reduce the computation time. However, compared with other methods, the computation speed is not fast enough. In the future, we will try to combine it with adaptive algorithms to further improve the computation efficiency. For problems that require long-time simulations, one may consider employing the time-marching method [31] to reduce computational costs. Similarly, for issues involving extensive spatial domains, regional partitioning can be explored to achieve computational efficiency.

6. Conclusions

This paper develops a space-time generalized finite difference method for the time fractional mobile/immobile diffusion equations. Time and space are considered together as a high-dimensional problem, and the integrand in the time fractional derivative term is approximated using linear interpolation, which transforms the equation into one composed of a linear combination of first-order time derivatives and second-order spatial derivatives. Then, according to the generalized finite difference method, derivatives are approximated by linear combinations of function values at neighboring nodes, which transforms the equation into a system of linear equations with sparse matrices of coefficients. Finally, through numerical simulations of one- and two-dimensional problems with sufficiently smooth solutions or with weak singularities at the initial moment, we

verify that the space-time generalized finite difference method is an accurate and robust local meshless algorithm, which has strong applicability in different problems.

It is noteworthy that further simulations and optimizations are also crucial for exploring the efficiency and accuracy of the proposed method. The space-time generalized finite difference method for solving fractional equations to obtain higher convergence order, on the one hand, increases the number of terms in the Taylor series expansion; on the other hand, requires the integrand in the fractional derivative to have more accurate approximation. In addition, in the fractional equations, the solution is usually not smooth enough at the initial moment due to the weak singularity of the fractional derivatives, necessitating appropriate adjustments when employing the space-time generalized finite difference method. In future work, we will apply this method to more complex models involving time, space, and time-space fractional derivatives, simulating non-Fickian diffusion problems in real-world, irregular, and anisotropic media. In addition, theoretical analysis of the number of nodes in the point star on the error will be a part of our future work.

Use of AI tools declaration

The author declares that she has not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported by the Shandong Provincial Natural Science Foundation, Grant ZR2022QA038, and Doctoral Research Foundation of Liaocheng University, Grant 318052155.

Conflicts of interest

The author declares that she has no competing interests.

References

1. R. Schumer, D. A. Benson, M. M. Meerschaert, B. Baeumer, Fractal mobile/immobile solute transport, *Water. Resour. Res.*, **39** (2003), 1296. https://doi.org/10.1029/2003WR002141
2. Y. Zhang, D. A. Benson, D. M. Reeves, Time and space nonlocalities underlying fractional-derivative models: Distinction and literature review of field applications, *Adv. Water. Resour.*, **32** (2009), 561–581. https://doi.org/10.1016/j.advwatres.2009.01.008
3. D. A. Benson, M. M. Meerschaert, A simple and efficient random walk solution of multi-rate mobile/immobile mass transport equations, *Adv. Water. Resour.*, **32** (2009), 532–539. https://doi.org/10.1016/j.advwatres.2009.01.002
4. S. K. Hansen, B. Berkowitz, Modeling non-Fickian solute transport due to mass transfer and physical heterogeneity on arbitrary groundwater velocity fields, *Water. Resour. Res.*, **56** (2020), e2019WR026868. https://doi.org/10.1029/2019WR026868

5. Y. Zhang, D. Zhou, M. Yin, H. Sun, W. Wei, S. Li, et al., Nonlocal transport models for capturing solute transport in one-dimensional sand columns: Model review, applicability, limitations and improvement, *Hydrol. Processes*, **34** (2020), 5104–5122. https://doi.org/10.1002/hyp.13930

6. Z. Zheng, Y. Wang, An averaged L1-type compact difference method for time-fractional mobile/immobile diffusion equations with weakly singular solutions, *Appl. Math. Lett.*, **131** (2022), 108076. https://doi.org/10.1016/j.aml.2022.108076

7. M. Fardi, M. Ghasemi, A numerical solution strategy based on error analysis for time-fractional mobile/immobile transport model, *Soft. Comput.*, **25** (2021), 11307–11331. https://doi.org/10.1007/s00500-021-05914-y

8. H. Zhang, X. Jiang, F. Liu, Error analysis of nonlinear time fractional mobile/immobile advection-diffusion equation with weakly singular solutions, *Fract. Calc. Appl. Anal.*, **24** (2021), 202–224. https://doi.org/10.1515/fca-2021-0009

9. O. Nikan, J. T. Machado, A. Golbabai, T. Nikazad, Numerical approach for modeling fractal mobile/immobile transport model in porous and fractured media, *Int. Commun. Heat. Mass.*, **111** (2020), 104443. https://doi.org/10.1016/j.icheatmasstransfer.2019.104443

10. J. Zhao, Z. Fang, H. Li, Y. Liu, Finite volume element method with the WSGD formula for nonlinear fractional mobile/immobile transport equations, *Adv. Differ. Equ.*, **2020** (2020), 1–20. https://doi.org/10.1186/s13662-020-02786-8

11. H. Jiang, D. Xu, W. Qiu, J. Zhou, An ADI compact difference scheme for the two-dimensional semilinear time-fractional mobile-immobile equation, *Comput. Appl. Math.*, **39** (2020), 1–17. https://doi.org/10.1007/s40314-020-01345-x

12. W. Qiu, D. Xu, J. Guo, J. Zhou, A time two-grid algorithm based on finite difference method for the two-dimensional nonlinear time-fractional mobile/immobile transport model, *Numer. Algor.*, **85** (2020), 39–58. https://doi.org/10.1007/s11075-019-00801-y

13. Q. Qin, L. Song, F. Liu, A meshless method based on the generalized finite difference method for three-dimensional elliptic interface problems, *Comput. Math. Appl.*, **131** (2023), 26–34. https://doi.org/10.1016/j.camwa.2022.11.020

14. Y. Gu, W. Qu, W. Chen, L. Song, C. Zhang, The generalized finite difference method for long-time dynamic modeling of three-dimensional coupled thermoelasticity problems, *J. Comput. Phys.*, **384** (2019), 42–59. https://doi.org/10.1016/j.jcp.2019.01.027

15. Y. Gu, J. Lei, C. Fan, X. He, The generalized finite difference method for an inverse time-dependent source problem associated with three-dimensional heat equation, *Eng. Anal. Bound. Elem.*, **91** (2018), 73–81. https://doi.org/10.1016/j.enganabound.2018.03.013

16. L. Song, P. Li, Y. Gu, C. Fan, Generalized finite difference method for solving stationary 2D and 3D Stokes equations with a mixed boundary condition, *Comput. Math. Appl.*, **80** (2020), 1726–1743. https://doi.org/10.1016/j.camwa.2020.08.004

17. F. Ureña, L. Gavete, A. Garcia, J. J. Benito, A. M. Vargas, Solving second order non-linear parabolic PDEs using generalized finite difference method (GFDM), *J. Comput. Appl. Math.*, **354** (2019), 221–241. https://doi.org/10.1016/j.cam.2018.02.016

18. X. Rao, H. Zhao, Y. Liu, A novel meshless method based on the virtual construction of node control domains for porous flow problems, *Eng. Comput.*, **40** (2024), 171–211. https://doi.org/10.1007/s00366-022-01776-6

19. W. Zhan, H. Zhao, X. Rao, Y. Liu, Generalized finite difference method-based numerical modeling of oil-water two-phase flow in anisotropic porous media, *Phys. Fluids*, **35** (2023), 103317. https://doi.org/10.1063/5.0166530

20. X. Rao, Y. Liu, H. Zhao, An upwind generalized finite difference method for meshless solution of two-phase porous flow equations, *Eng. Anal. Bound. Elem.*, **137** (2022), 105–118. https://doi.org/10.1016/j.enganabound.2022.01.013

21. H. G. Sun, Z. Wang, J. Nie, Y. Zhang, R. Xiao, Generalized finite difference method for a class of multidimensional space-fractional diffusion equations, *Comput. Mech.*, **67** (2021), 17–32. https://doi.org/10.1007/s00466-020-01917-y

22. W. Hu, Z. Fu, Z. Tang, Y. Gu, A meshless collocation method for solving the inverse cauchy problem associated with the variable-order fractional heat conduction model under functionally graded materials, *Eng. Anal. Bound. Elem.*, **140** (2022), 132–144. https://doi.org/10.1016/j.enganabound.2022.04.007

23. V. R. Hosseini, A. A. Mehrizi, H. Karimi-Maleh, M. Naddafi, A numerical solution of fractional reaction-convection-diffusion for modeling PEM fuel cells based on a meshless approach, *Eng. Anal. Bound. Elem.*, **155** (2023), 707–716. https://doi.org/10.1016/j.enganabound.2023.06.016

24. W. Z. Qu, H. He, A spatial-temporal GFDM with an additional condition for transient heat conduction analysis of FGMs, *Appl. Math. Lett.*, **110** (2020), 106579. https://doi.org/10.1016/j.aml.2020.106579

25. P. Li, Space-time generalized finite difference nonlinear model for solving unsteady Burger' equations, *Appl. Math. Lett.*, **114** (2021), 106896. https://doi.org/10.1016/j.aml.2020.106896

26. F. Liu, L. Song, M. Jiang, Space-time generalized finite difference method for solving the thin elastic plate bending under dynamic loading, *Eng. Anal. Bound. Elem.*, **143** (2022), 632–638. https://doi.org/10.1016/j.enganabound.2022.07.015

27. F. Zhang, P. Li, Y. Gu, C. Fan, A space-time generalized finite difference scheme for long wave propagation based on high-order Korteweg-de Vries type equations, *Math. Comput. Simulat.*, **228** (2025), 298–312. https://doi.org/10.1016/j.matcom.2024.09.012

28. J. Benito, A. García, M. Negreanu, F. Ureña, A. Vargas, Two finite difference methods for solving the Zakharov-Kuznetsov-Modified Equal-Width equation, *Eng. Anal. Bound. Elem.*, **153** (2023), 213–225. https://doi.org/10.1016/j.enganabound.2023.05.003

29. P. Li, S. Hu, M. Zhang, Numerical solutions of the nonlinear dispersive shallow water wave equations based on the space–time coupled generalized finite difference scheme, *Appl. Sci.*, **13** (2023), 8504. https://doi.org/10.3390/app13148504

30. J. J. Benito, Á. García, M. Negreanu, F. Ureña, A. M. Vargas, A novel spatio-temporal fully meshless method for parabolic PDEs, *Mathematics*, **10** (2022), 1870. https://doi.org/10.3390/math10111870

31. P. Li, J. K. Grabski, C. Fan, F. Wang, A space-time generalized finite difference method for solving unsteady double-diffusive natural convection in fluid-saturated porous media, *Eng. Anal. Bound. Elem.*, **142** (2022), 138–152. https://doi.org/10.1016/j.enganabound.2022.04.038

32. J. Benito, F. Urena, L. Gavete, Influence of several factors in the generalized finite difference method, *Appl. Math. Model.*, **25** (2001), 1039–1053. https://doi.org/10.1016/S0307-904X(01)00029-4

33. C. Fan, P. Li, Generalized finite difference method for solving two-dimensional Burgers' equations, *Procedia Eng.*, **79** (2014), 55–60. https://doi.org/10.1016/j.proeng.2014.06.310

34. H. Qiao, A. Cheng, A fast finite difference method for 2D time fractional mobile/immobile equation with weakly singular solution, *Fractal. Fract.*, **9** (2025), 204. https://doi.org/10.3390/fractalfract9040204

AIMS Press