Accepted Manuscript

An unstructured mesh control volume method for two-dimensional space fractional diffusion equations with variable coefficients on convex domains

Libo Feng, Fawang Liu, Ian Turner

PII: S0377-0427(19)30316-4
DOI: https://doi.org/10.1016/j.cam.2019.06.035
Reference: CAM 12319

To appear in: *Journal of Computational and Applied Mathematics*

Received date : 30 March 2018
Revised date : 10 April 2019

Highlights

1. A novel unstructured mesh control volume method to deal with the space fractional derivative on arbitrarily shaped convex domains is proposed.
2. The finite volume scheme for a 2D space fractional diffusion equation with variable coefficients is presented.
3. A fast iterative method is developed.
4. Compared to the FEM, the FVM reduce CPU time significantly.
5. The FVM can be applied to problems on arbitrarily shaped convex domains.

An unstructured mesh control volume method for two-dimensional space fractional diffusion equations with variable coefficients on convex domains

Libo Feng[a], Fawang Liu[a,b,*], Ian Turner[a,c]

[a]School of Mathematical Sciences, Queensland University of Technology, GPO Box 2434, Brisbane, Qld 4001, Australia
[b]College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350116, China.
[c]Australian Research Council Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS), Queensland University of Technology (QUT), Brisbane, Australia.

Abstract

In this paper, we propose a novel unstructured mesh control volume method to deal with the space fractional derivative on arbitrarily shaped convex domains, which to the best of our knowledge is a new contribution to the literature. Firstly, we present the finite volume scheme for the two-dimensional space fractional diffusion equation with variable coefficients and provide the full implementation details for the case where the background interpolation mesh is based on triangular elements. Secondly, we explore the property of the stiffness matrix generated by the integral of the space fractional derivative. We find that the stiffness matrix is sparse and not regular. Therefore, we choose a suitable sparse storage format for the stiffness matrix and develop a fast iterative method to solve the linear system, which is more efficient than using the Gaussian elimination method. Finally, we present several examples to verify our method, in which we make a comparison of our method with the finite element method for solving a Riesz space fractional diffusion equation on a circular domain. The numerical results demonstrate that our method can reduce CPU time significantly while retaining the same accuracy and approximation property as the finite element method. The numerical results also illustrate that our method is effective and reliable and can be applied to problems on arbitrarily shaped convex domains.

Keywords: control volume method, unstructured mesh, fast iterative solver, space fractional derivative, irregular convex domains, two-dimensional

1. Introduction

In the past two decades, fractional differential equations have been applied in many fields of science [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12], in which space fractional diffusion equations are used to model the anomalous transport of solute in groundwater hydrology [13, 14]. For space fractional diffusion equations with constant coefficients, analytical solutions can be obtained by utilizing the Fourier transform methods. However, many practical problems involve variable coefficients [15, 16], in which the diffusion velocity can vary over the solution domain. The work involving space fractional diffusion equations with variable coefficients is numerous. Meerschaert et al. [13, 17] considered the finite difference method for the one-dimensional one-sided and two-sided space fractional diffusion equations with variable coefficients, respectively. Zhang et al. [18] explored the homogeneous space-fractional advection-dispersion equation with space-dependent coefficients. Ding et al. [19] presented the weighted finite difference methods for a class of space fractional partial differential equations with variable coefficients. Moroney and Yang [20, 21] proposed some fast preconditioners for the numerical solution of a class of two-sided nonlinear space-fractional diffusion equations with variable coefficients. Chen and Deng [22] discussed the alternating direction implicit method to solve a two-dimensional, two-sided space fractional convection-diffusion equation on a finite domain. Wang and Zhang [23] developed a high-accuracy preserving spectral Galerkin method for the Dirichlet boundary-value problem of a one-sided variable-coefficient conservative fractional diffusion equation. Liu et al. [24] developed a new fractional finite volume method for solving the fractional diffusion equation with with a spaceCtime dependent variable coefficien. Li et al. [25, 26] developed novel finite volume methods for Riesz space

*Corresponding author.
Email address: f.liu@qut.edu.au (Fawang Liu)

distributed-order diffusion equation and the Riesz space distributed- order advection-diffusion equation. Feng et al. [27] proposed the finite volume method for a two-sided space-fractional diffusion equation with variable coefficients. Chen et al. [28] considered an inverse problem for identifying the fractional derivative indices in a two-dimensional space-fractional nonlocal model with variable diffusivity coefficients. Jia and Wang [29] presented a fast finite volume method for conservative space-fractional diffusion equations with variable coefficients. In [30], Feng et al. presented a new second order finite difference scheme for a two-sided space-fractional diffusion equation with variable coefficients. Chen et al. [31] presented numerical methods and analysis for a multi-term time-space variable-order fractional advection-diffusion equations and applications. Liu et al. [32] proposed numerical methods for solving the multi-term time fractional wave equations.

In fact, many mathematical models and problems from science and engineering must be computed on irregular domains and therefore seeking effective numerical methods to solve these problems on such domains is important. Although existing numerical methods for fractional diffusion equations are numerous [33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43], most of them are limited to regular domains and uniform meshes. Research involving unstructured meshes and irregular domains is sparse. Liu et al. [44] presented unstructured-mesh Galerkin finite element method for the two-dimensional multi-term timeCspace fractional Bloch-Torrey equations on irregular convex domains. Fan et al. [45] presented unstructured mesh finite element method for the two-dimensional multi-term time-space fractional diffusion-wave equation on an irregular convex domain. Yang et al. [46] proposed the finite volume scheme for a two-dimensional space-fractional reaction-diffusion equation based on the fractional Laplacian operator $-(-\nabla^2)^{\frac{\alpha}{2}}$, which was computed using unstructured triangular meshes on a unit disk. Burrage et al. [47] developed some techniques for solving fractional-in-space reaction diffusion equations using the finite element method on both structured and unstructured grids. Qiu et al. [48] developed the nodal discontinuous Galerkin method for fractional diffusion equations on a two-dimensional domain with triangular meshes. Liu et al. [49] presented the semi-alternating direction method for a two-dimensional fractional FitzHugh-Nagumo monodomain model on an approximate irregular domain. Qin et al. [50] also used the implicit alternating direction method to solve a two-dimensional fractional Bloch-Torrey equation using an approximate irregular domain. Karaa et al. [51] proposed a finite volume element method implemented on an unstructured mesh for approximating the anomalous subdiffusion equations with a temporal fractional derivative. Yang et al. [52] established the unstructured mesh finite element method for the nonlinear Riesz space fractional diffusion equations on irregular convex domains. Fan et al. [53] extended the unstructured mesh finite element method developed by Yang et al. [52] to the time-space fractional wave equation. Feng et al. [54] investigated the unstructured mesh finite element method for a two-dimensional time-space Riesz fractional diffusion equation on irregular arbitrarily shaped convex domains and a multiply-connected domain. Le et al. [55] studied the finite element approximation for a time-fractional diffusion problem on a domain with a re-entrant corner. To the best of our knowledge, the control volume finite element method (see Carr et al. [56] for an illustration of the method applied to wood drying) has not been generalised to allow the solution of space fractional diffusion equations with variable coefficients.

In this paper, we will consider the unstructured mesh control volume method for the following two-dimensional space fractional diffusion equation with variable coefficients (2D SFDE-VC) [28] on an arbitrarily shaped convex domain:

$$\begin{aligned}\frac{\partial u(x,y,t)}{\partial t} =& \frac{\partial}{\partial x}\left[K_1(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial x^\alpha} - K_2(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial(-x)^\alpha}\right] \\ &+\frac{\partial}{\partial y}\left[K_3(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial y^\beta} - K_4(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial(-y)^\beta}\right] \\ &+f(x,y,t), \quad (x,y,t)\in\Omega\times(0,T], \end{aligned} \tag{1}$$

subject to the initial condition

$$u(x,y,0) = \phi(x,y), \quad (x,y)\in\overline{\Omega}, \tag{2}$$

and boundary conditions

$$u(x,y,t) = 0, \quad (x,y,t)\in\partial\Omega\times[0,T], \tag{3}$$

where $0<\alpha,\ \beta<1$, $K_i(x,y,t)\geq 0$, $i=1,2,3,4$, $f(x,y,t)$ and $\phi(x,y)$ are assumed to be two known smooth functions. When the solution domain is rectangular $\Omega=(a,b)\times(c,d)$, we define the Riemman-Liouville fractional derivative as [57]:

$$
\begin{aligned}
\frac{\partial^{\alpha}u(x,y,t)}{\partial x^{\alpha}} &= {}_{a}D_{x}^{\alpha}u(x,y,t) = \frac{1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\int_{a}^{x}(x-s)^{-\alpha}u(s,y,t)\ ds,\\
\frac{\partial^{\alpha}u(x,y,t)}{\partial(-x)^{\alpha}} &= {}_{x}D_{b}^{\alpha}u(x,y,t) = \frac{-1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\int_{x}^{b}(s-x)^{-\alpha}u(s,y,t)\ ds,\\
\frac{\partial^{\beta}u(x,y,t)}{\partial y^{\beta}} &= {}_{c}D_{y}^{\beta}u(x,y,t) = \frac{1}{\Gamma(1-\beta)}\frac{\partial}{\partial y}\int_{c}^{y}(y-s)^{-\beta}u(x,s,t)\ ds,\\
\frac{\partial^{\beta}u(x,y,t)}{\partial(-y)^{\beta}} &= {}_{y}D_{d}^{\beta}u(x,y,t) = \frac{-1}{\Gamma(1-\beta)}\frac{\partial}{\partial y}\int_{y}^{d}(s-y)^{-\beta}u(x,s,t)\ ds.
\end{aligned}
$$

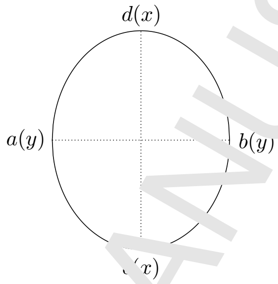

Figure 1: The illustration of a solution domain with curved boundary

When the boundary of the solution domain is non-constant or curved, for example a convex domain shown in Figure 1 with left boundary $a(y)$, right boundary $b(y)$, lower boundary $c(x)$ and upper boundary $d(x)$, we define the Riemman-Liouville fractional derivative as [54]:

$$
\begin{aligned}
\frac{\partial^{\alpha}u(x,y,t)}{\partial x^{\alpha}} &= {}_{a(y)}D_{x}^{\alpha}u(x,y,t) = \frac{1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\int_{a(y)}^{x}(x-s)^{-\alpha}u(s,y,t)\ ds,\\
\frac{\partial^{\alpha}u(x,y,t)}{\partial(-x)^{\alpha}} &= {}_{x}D_{b(y)}^{\alpha}u(x,y,t) = \frac{-1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\int_{x}^{b(y)}(s-x)^{-\alpha}u(s,y,t)\ ds,\\
\frac{\partial^{\beta}u(x,y,t)}{\partial y^{\beta}} &= {}_{c(x)}D_{y}^{\beta}u(x,y,t) = \frac{1}{\Gamma(1-\beta)}\frac{\partial}{\partial y}\int_{c(x)}^{y}(y-s)^{-\beta}u(x,s,t)\ ds,\\
\frac{\partial^{\beta}u(x,y,t)}{\partial(-y)^{\beta}} &= {}_{y}D_{d(x)}^{\beta}u(x,y,t) = \frac{-1}{\Gamma(1-\beta)}\frac{\partial}{\partial y}\int_{y}^{d(x)}(s-y)^{-\beta}u(x,s,t)\ ds.
\end{aligned}
$$

Remark 1.1. *When $K_i(x,y,t), i = 1,2,3,4$ take the special form*

$$
\begin{aligned}
K_1(x,y,t) = K_2(x,y,t) &= -\frac{K_x}{2\cos\frac{\pi(1+\alpha)}{2}},\\
K_3(x,y,t) = K_4(x,y,t) &= -\frac{K_y}{2\cos\frac{\pi(1+\beta)}{2}},
\end{aligned}
$$

equation (1) can be written as the following Riesz space fractional diffusion equation [49, 52]

$$
\frac{\partial u(x,y,t)}{\partial t} = K_x\frac{\partial^{1+\alpha}u(x,y,t)}{\partial|x|^{1+\alpha}} + K_y\frac{\partial^{1+\beta}u(x,y,t)}{\partial|y|^{1+\beta}} + f(x,y,t), \tag{4}
$$

where

$$\frac{\partial^{1+\alpha}u(x,y,t)}{\partial|x|^{1+\alpha}} = -\frac{1}{2\cos\frac{\pi(1+\alpha)}{2}}\left[\frac{\partial^{1+\alpha}u(x,y,t)}{\partial x^{1+\alpha}} + \frac{\partial^{1+\alpha}u(x,y,t)}{\partial(-x)^{1+\alpha}}\right],$$

$$\frac{\partial^{1+\beta}u(x,y,t)}{\partial|y|^{1+\beta}} = -\frac{1}{2\cos\frac{\pi(1+\beta)}{2}}\left[\frac{\partial^{1+\beta}u(x,y,t)}{\partial y^{1+\beta}} + \frac{\partial^{1+\beta}u(x,y,t)}{\partial(-y)^{1+\beta}}\right].$$

One important application of equation (4) is in the study of cardiac arrhythmias. In two dimensions, the fractional FitzHugh-Nagumo monodomain model can be rewritten as a two-dimensional Riesz space fractional reaction-diffusion model, which can be used to describe the propagation of the electrical potential in heterogeneous cardiac tissue [49, 58]. This electrophysiological model of the heart can describe how electrical currents flow through the heart controlling its contraction and can be used to ascertain the effects of certain drugs designed to treat heart problems.

The major contribution of this paper is as follows.

- Different from [46] and [51], we consider the control volume method for the two-dimensional space fractional diffusion equation with variable coefficients, in which the space fractional operator is either the Riemman-Liouville fractional derivative or Riesz space fractional derivative. To the best of our knowledge, this is a new contribution to the literature.

- We propose a novel technique utilizing the control volume method implemented with an unstructured triangular mesh to deal with the space fractional derivative on an irregular convex domain, which we believe provides a very flexible solution strategy because our considered solution domain can be arbitrarily convex. Compared to the finite difference method in [49, 56], our method requires fewer grid nodes to generate the meshes in the solution domain partition.

- For the methods considered in this paper, we construct the control volumes using triangular meshes and transform the problem (1) from the solution domain to a single control volume. Then we integrate problem (1) over an arbitrary control volume and change the control volume integral to a line integral over the control volume faces, which is approximated by the midpoint approximation. Moreover, we utilise the linear basis function to approximate the fractional derivatives at the midpoints of the control volume faces, in which some numerical techniques are used to handle the non-locality of the fractional derivative of the basis function.

- We explore the property of the stiffness matrix generated by the integral of the space fractional derivative. We find that the stiffness matrix is sparse and not regular. Especially, the smaller the maximum edge of the triangulation is, the more sparse of the stiffness matrix becomes. Therefore, we choose a suitable sparse storage format for the stiffness matrix and utilise the bi-conjugate gradient stabilized method (Bi-CGSTAB) iterative method to solve the linear system, which is more efficient than using the Gaussian elimination method.

- We present several examples to verify our method, in which we make a comparison of our method with the finite element method proposed in [52] for solving the Riesz space fractional diffusion equation (4) on a circular domain. In [52], the authors develop an algorithm to form the stiffness matrix and derive the bilinear operator as

$$A(u,v) = \frac{K_x}{2\cos\frac{\pi(1+\alpha)}{2}}\left\{\left({}_{a(y)}D_x^{\frac{(1+\alpha)}{2}}u, {}_xD_{b(y)}^{\frac{(1+\alpha)}{2}}v\right) + \left({}_xD_{b(y)}^{\frac{(1+\alpha)}{2}}u, {}_{a(y)}D_x^{\frac{(1+\alpha)}{2}}v\right)\right\}$$
$$+ \frac{K_y}{2\cos\frac{\pi(1+\beta)}{2}}\left\{\left({}_{c(x)}D_y^{\frac{(1+\beta)}{2}}u, {}_yD_{d(x)}^{\frac{(1+\beta)}{2}}v\right) + \left({}_yD_{d(x)}^{\frac{(1+\beta)}{2}}u, {}_{c(x)}D_y^{\frac{(1+\beta)}{2}}v\right)\right\}.$$

The bilinear form involves eight fractional derivative terms and the approximation of two-fold multiple integrals, which are approximated by Gauss quadrature. While for the control volume method, we use the following form to generate the stiffness matrix form,

$$\frac{K_x}{2\cos\frac{\pi(1+\alpha)}{2}}\oint_{\Gamma_i}\left[\frac{\partial^{\alpha}u(x,y,t)}{\partial x^{\alpha}} - \frac{\partial^{\alpha}u(x,y,t)}{\partial(-x)^{\alpha}}\right]dy$$
$$-\frac{K_y}{2\cos\frac{\pi(1+\beta)}{2}}\oint_{\Gamma_i}\left[\frac{\partial^{\beta}u(x,y,t)}{\partial y^{\beta}} - \frac{\partial^{\beta}u(x,y,t)}{\partial(-y)^{\beta}}\right]dx,$$

in which we only need to calculate 4 fractional derivative terms and the approximation of line integrals. The numerical results demonstrate that our method can reduce CPU time significantly while retaining the same accuracy and approximation property as the finite element method. The numerical results also illustrate that our method is effective and reliable and can be applied to problems on arbitrarily convex domains.

The outline of this paper is as follows. In section 2, the unstructured mesh control volume method for the problem (1) is proposed and the full implementation details are provided. Then the property of the stiffness matrix is explored and a fast iterative solver is developed for the linear system. In section 3, several numerical examples are presented to verify the effectiveness of the method and comparisons are made with existing methods to highlight its computational performance. Finally, some conclusions of the work are drawn.

2. Control volume finite element method

In this section, we will generalise the control volume method to solve equation (1), placing particular emphasis on the way the Riemman-Liouville fractional derivatives are discretised in space. Firstly, we divide the solution domain Ω into a number of regular triangular regions. Let $\mathcal{T}_h$ denote this triangulation and h be the maximum diameter of the triangular elements. Then we introduce the control volumes, which are constructed as follows. Let M_h be a set of vertices,

$$M_h = \{P_i : \ P_i \text{ is a vertex of the element } K \in \mathcal{T}_h \text{ and } P_i \in \Omega\},$$

and M_h^0 be the set of interior nodes in $\mathcal{T}_h$. We denote P_0 as the interior node of the triangulation $\mathcal{T}_h$ and P_i $(i = 1, 2, \cdots, m)$ as its adjacent nodes (see Figure 2 with $m = 6$). Let S_i $(i = 1, 2, \cdots, m)$ be the midpoints of the line segments $\overline{P_0P_i}$ and Q_i $(i = 1, 2, \cdots, m)$ the barycenters of the triangle $\Delta P_0P_iP_{i+1}$ with $P_{m+1} = P_1$. The control volume $K^*_{P_0}$ is constructed by joining successively $S_1, \ Q_1, \ \cdots, \ S_m, \ Q_m, \ S_1$ (see Figure 2). We call the line segments $\overline{S_iQ_i}$ and $\overline{Q_iS_{i+1}}$ $(i = 1, 2, \cdots, m$ and $S_{m+1} = S_1)$ control volume faces. Consequently, each of the triangular elements is divided into three sub-domains by these control surfaces. These quadrilateral shapes are called sub-control volumes and are illustrated in Figure 2 (for example, the quadrilateral $S_1Q_1S_2P_0$). Thus, a control volume consists of the sum of all neighbouring sub-control volumes that surround the given node P_0. The control volume is polygonal in shape and can be assembled in a straightforward and efficient manner at the element level. The flow across each control surface must be determined by an integral. Therefore, the finite volume method discretization process is initiated by utilising the integrated form of equation (1).

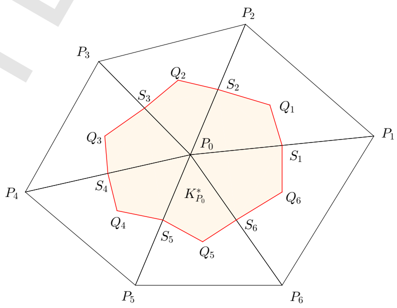

Figure 2: The illustration of a control volume

Integrating (1) over an arbitrary control volume V_i $(i = 1, 2, \cdots, N_p)$, yields

$$\begin{aligned}\int_{V_i} \frac{\partial u(x,y,t)}{\partial t}\, dV_i = &\int_{V_i} \frac{\partial}{\partial x}\left[K_1(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial x^\alpha} - K_2(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial(-x)^\alpha}\right] dV_i \\ &+ \int_{V_i} \frac{\partial}{\partial y}\left[K_3(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial y^\beta} - K_4(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial(-y)^\beta}\right] dV_i \\ &+ \int_{V_i} f(x,y,t)\, dV_i. \end{aligned} \tag{5}$$

Utilising a lumped mass approach for the time derivative and source term and applying Green's theorem to the other two integral terms, gives

$$\begin{aligned}\Delta V_i \frac{\partial u(x,y,t)}{\partial t}\bigg|_{(x_i,y_i)} &= \oint_{\Gamma_i}\left[K_1(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial x^\alpha} - K_2(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial(-x)^\alpha}\right]dy \\ &- \oint_{\Gamma_i}\left[K_3(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial y^\beta} - K_4(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial(-y)^\beta}\right]dx \\ &+\Delta V_i f(x_i,y_i,t), \end{aligned} \tag{6}$$

where Γ_i is the boundary of control volume V_i. We assume the finite volume integration is an anticlockwise traversal and the outward unit normal surface vector to the control surface is shown in Figure 3 with $\Delta x = x_b - x_a$ and $\Delta y = y_b - y_a$. Denote ΔV_i and ΔV_{ij} the area of the control volume and the sub-control volume surrounding the

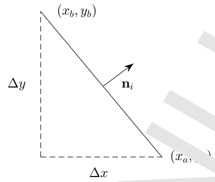

Figure 3: A control volume face and the outward normal unit vector

point (x_i, y_i), then we have

$$\Delta V_i = \sum_{j=1}^{m_i} \Delta V_{ij},$$

where m_i is the total number of sub-control volumes that make up the control volume associated with the node i. The integral term on the right-hand side of equation (1) is a line integral, which can be approximated by the midpoint approximation for each control surface. Hence, the first integral term in equation (6) can be rewritten as

$$\begin{aligned}&\oint_{\Gamma_i}\left[K_1(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial x^\alpha} - K_2(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial(-x)^\alpha}\right]dy \\ =&\sum_{j=1}^{m_i}\sum_{r=1}^{2}\left[K_1(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial x^\alpha} - K_2(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial(-x)^\alpha}\right]\bigg|_{(x_r,y_r)} \Delta y^i_{j,r}, \end{aligned} \tag{7}$$

where (x_r, y_r) is the mid-point of the control face (CF) (see Figure 4). Similarly, for the second integral term in

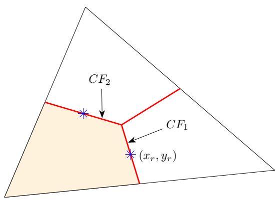

Figure 4: The illustration of control faces with mid-points

equation (6), we have

$$\begin{aligned}&\oint_{\Gamma_i}\left[K_3(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial y^\beta} - K_4(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial(-y)^\beta}\right]dx \\ =&\sum_{j=1}^{m_i}\sum_{r=1}^{2}\left[K_3(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial y^\beta} - K_4(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial(-y)^\beta}\right]\bigg|_{(x_r,y_r)} \Delta x^i_{j,r}. \end{aligned} \tag{8}$$

Substituting equations (7) and (8) into (6), we obtain

$$
\begin{aligned}
&\Delta V_i \frac{\partial u(x,y,t)}{\partial t}\Big|_{(x_i,y_i)} \\
&= \sum_{j=1}^{m_i}\sum_{r=1}^{2}\left[K_1(x,y,t)\frac{\partial^{\alpha}u(x,y,t)}{\partial x^{\alpha}} - K_2(x,y,t)\frac{\partial^{\alpha}u(x,y,t)}{\partial(-x)^{\alpha}}\right]\Big|_{(x_r,y_r)}\Delta y^i_{j,r} \\
&-\sum_{j=1}^{m_i}\sum_{r=1}^{2}\left[K_3(x,y,t)\frac{\partial^{\beta}u(x,y,t)}{\partial y^{\beta}} - K_4(x,y,t)\frac{\partial^{\beta}u(x,y,t)}{\partial(-y)^{\beta}}\right]\Big|_{(x_r,y_r)}\Delta x^i_{j,r} \\
&+\Delta V_i f(x_i,y_i,t).
\end{aligned}
\tag{9}
$$

To discretise the time derivative in equation (9) at $t = t_n$, we use the backward Euler difference scheme

$$
\frac{\partial u(x,y,t_n)}{\partial t} = \frac{u(x,y,t_n) - u(x,y,t_{n-1})}{\tau} + O(\tau). \tag{10}
$$

In the following, we discuss the spatial discretisation of $u(x,y,t_n)$. We consider the computation process for piecewise linear polynomials on the triangular element e_p, $p = 1, 2, \ldots, N_e$, where N_e is the total number of triangles. Then, within element e_p, the field function $u^p(x,y)$ can be written as

$$
u^p(x,y) = \sum_{j=1}^{3} u_j\ \varphi_j(x,y) + O(h^2),
$$

where the triangle vertices are numbered in a counter-clockwise order as $1,\ 2,\ 3$ and the basis function $\varphi_j(x,y)$ is defined as

$$
\begin{aligned}
&\varphi_j(x,y)\Big|_{(x,y)\in e_p} = \frac{1}{2\Delta_{e_p}}(a_j\,x + b_j\ y + c_j), \quad \varphi_j(x,y)\Big|_{(x,y)\notin e_p} = 0,\\
&a_1 = y_2 - y_3,\ a_2 = y_3 - y_1,\ a_3 = y_1 - y_2,\\
&b_1 = x_3 - x_2,\ b_2 = x_1 - x_3,\ b_3 = x_2 - x_1,\\
&c_1 = x_2y_3 - x_3y_2,\ c_2 = x_3y_1 - x_1y_3,\ c_3 = x_1y_2 - x_2y_1,
\end{aligned}
$$

where Δ_{e_p} is the area of triangle element p. It is well-known that

$$
\varphi_j(x_i,y_i) = \delta_{ij}, \quad i,\ j = 1,\ 2,\ 3,
$$

where δ is the Kronecker function. With these local field functions and basis functions, we can obtain a global approximation of $u(x,y)$ for the whole triangulation:

$$
u(x,y) = \sum_{k=1}^{N_p} u_k\ l_k(x,y) + O(h^2),
$$

where $l_k(x,y)$ is the new basis function whose support domain is Ω_{e_k} (see Figure 5 the green polygonal domain) and N_p is the total number of vertices on the convex domain Ω.

Now, we denote $u_h(x,y,t_n)$ as the approximation solution of $u(x,y,t_n)$ and write $u_h(x,y,t_n)$ in the form

$$
u_h(x,y,t_n) = \sum_{k=1}^{N_p} u_k^n\ l_k(x,y), \tag{11}
$$

where u_k^n are the coefficients that are to be solved for. Substituting equations (10) and (11) into equation (9), we discretise equation (9) at $t = t_n$ as follows:

$$\begin{aligned}
&\Delta V_i \sum_{k=1}^{N_p} \frac{u_k^n - u_k^{n-1}}{\tau} l_k(x_i, y_i) \\
&= \sum_{k=1}^{N_p} \sum_{j=1}^{m_i} \sum_{r=1}^{2} u_k^n \Big[K_1(x,y,t) \frac{\partial^\alpha l_k(x,y)}{\partial x^\alpha} - K_2(x,y,t) \frac{\partial^\alpha l_k(x,y)}{\partial (-x)^\alpha} \Big]\Big|_{(x_r,y_r)} \Delta y_{j,r}^i \\
&- \sum_{k=1}^{N_p} \sum_{j=1}^{m_i} \sum_{r=1}^{2} u_k^n \Big[K_3(x,y,t) \frac{\partial^\beta l_k(x,y)}{\partial y^\beta} - K_4(x,y,t) \frac{\partial^\beta l_k(x,y)}{\partial (-y)^\beta} \Big]\Big|_{(x_r,y_r)} \Delta x_{j,r}^i \\
&+ \Delta V_i f(x_i, y_i, t_n). \qquad (12)
\end{aligned}$$

Using the fact that

$$l_k(x_i, y_i) = \begin{cases} 1, & i = k, \\ 0, & i \neq k, \end{cases}$$

we obtain

$$\begin{aligned}
&\Delta V_i \frac{u_i^n - u_i^{n-1}}{\tau} \\
&= \sum_{k=1}^{N_p} \sum_{j=1}^{m_i} \sum_{r=1}^{2} u_k^n \Big[K_1(x,y,t) \frac{\partial^\alpha l_k(x,y)}{\partial x^\alpha} - K_2(x,y,t) \frac{\partial^\alpha l_k(x,y)}{\partial (-x)^\alpha} \Big]\Big|_{(x_r,y_r)} \Delta y_{j,r}^i \\
&- \sum_{k=1}^{N_p} \sum_{j=1}^{m_i} \sum_{r=1}^{2} u_k^n \Big[K_3(x,y,t) \frac{\partial^\beta l_k(x,y)}{\partial y^\beta} - K_4(x,y,t) \frac{\partial^\beta l_k(x,y)}{\partial (-y)^\beta} \Big]\Big|_{(x_r,y_r)} \Delta x_{j,r}^i \\
&+ \Delta V_i f(x_i, y_i, t_n). \qquad (13)
\end{aligned}$$

Equation (13) can be written in the following matrix form

$$\mathbf{A} \frac{\mathbf{U}^n - \mathbf{U}^{n-1}}{\tau} = \mathbf{M}\mathbf{U}^n + \mathbf{A}\mathbf{F}^n, \qquad (14)$$

where $\mathbf{A} = \mathrm{diag}\,[\Delta V_1, \Delta V_2, \ldots, \Delta V_{N_p}]$, $\mathbf{U}^n = [u_1^n, u_2^n, \ldots, u_{N_p}^n]^T$, $\mathbf{F}^n = [f(x_1, y_1, t_n), f(x_2, y_2, t_n), \ldots, f(x_{N_p}, y_{N_p}, t_n)]^T$. Rearranging we obtain

$$(\mathbf{A} - \tau\mathbf{M})\mathbf{U}^n = \mathbf{A}\mathbf{U}^{n-1} + \tau\mathbf{A}\mathbf{F}^n. \qquad (15)$$

To form matrix $\mathbf{M}$, we need to calculate the fractional derivative of the basis function $l_k(x,y)$. In the following, we focus on the calculation of $\frac{\partial^\alpha l_k(x,y)}{\partial x^\alpha}$, $\frac{\partial^\alpha l_k(x,y)}{\partial (-x)^\alpha}$, $\frac{\partial^\beta l_k(x,y)}{\partial y^\beta}$ and $\frac{\partial^\beta l_k(x,y)}{\partial (-y)^\beta}$ at (x_r, y_r). To evaluate $\frac{\partial^\alpha l_k(x,y)}{\partial x^\alpha}\big|_{(x_r,y_r)}$ and $\frac{\partial^\alpha l_k(x,y)}{\partial (-x)^\alpha}\big|_{(x_r,y_r)}$, suppose that line $y = y_r$ intersects n_q points with the support domain Ω_{e_k} of $l_k(x,y)$ (see Figure 5 with $n_q = 5$).

Then we have

$$\begin{aligned}
\frac{\partial^\alpha l_k(x,y)}{\partial x^\alpha}\Big|_{(x_r,y_r)} &= \frac{\partial^\alpha l_k(x,y_r)}{\partial x^\alpha}\Big|_{x=x_r}, \\
\frac{\partial^\alpha l_k(x,y)}{\partial (-x)^\alpha}\Big|_{(x_r,y_r)} &= \frac{\partial^\alpha l_k(x,y_r)}{\partial (-x)^\alpha}\Big|_{x=x_r}.
\end{aligned}$$

Using the important observation that

$$l_k(x, y_r) = \begin{cases} 0, & a \le x \le x_1, \\ \varphi_{k4}(x, y_r), & x_1 \le x \le x_2, \\ \varphi_{k3}(x, y_r), & x_2 \le x \le x_3, \\ \varphi_{k2}(x, y_r), & x_3 \le x \le x_4, \\ \varphi_{k1}(x, y_r), & x_4 \le x \le x_5, \\ 0, & x_5 \le x \le b, \end{cases}$$

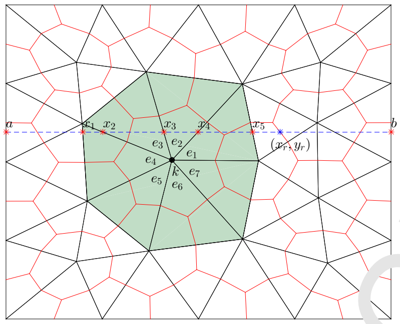

Figure 5: The illustration of line $y = y_r$ intersecting n_q points with the support domain Ω_{e_k} of $l_k(x, y)$, where (x_r, y_r) locates out of Ω_{e_k}.

where $\varphi_{kp}(x, y)$ is the basis function of node k on the triangular element e_p, we obtain

$$
\begin{aligned}
&\frac{\partial^\alpha l_k(x, y_r)}{\partial x^\alpha}\Big|_{x=x_r} \\
&= \left(\frac{1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\int_a^x (x-\xi)^{-\alpha} l_k(\xi, y_r)d\xi\right)\Big|_{x=x_r} \\
&= \left[\frac{1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\left(\int_a^{x_1} + \int_{x_1}^{x_2} + \int_{x_2}^{x_3} + \int_{x_3}^{x_4} + \int_{x_4}^{x_5} + \int_{x_5}^{x}\right)(x-\xi)^{-\alpha} l_k(\xi, y_r)d\xi\right]\Big|_{x=x_r} \\
&= \left[\frac{1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\left(\int_{x_1}^{x_2} + \int_{x_2}^{x_3} + \int_{x_3}^{x_4} + \int_{x_4}^{x_5}\right)(x-\xi)^{-\alpha} l_k(\xi, y_r)d\xi\right]\Big|_{x=x_r}.
\end{aligned} \tag{16}
$$

As $l_k(x, y_r)$ is a linear function on each sub integral interval, equation (16) can be evaluated using integration by parts over each sub integral interval. For the right fractional derivative of $l_k(x, y_r)$ at (x_r, y_r), we obtain

$$
\frac{\partial^\alpha l_k(x, y_r)}{\partial(-x)^\alpha}\Big|_{x=x_r} = \left(\frac{-1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\int_x^b (\xi - x)^{-\alpha} l_k(\xi, y_r)d\xi\right)\Big|_{x=x_r} = 0. \tag{17}
$$

Now we consider the case that point (x_r, y_r) is in the support domain Ω_{e_k} of $l_k(x, y)$. Suppose that line $y = y_r$ intersects n_q points with the support domain Ω_{e_k} (see Figure 6 with $n_q = 4$). In this case, we have

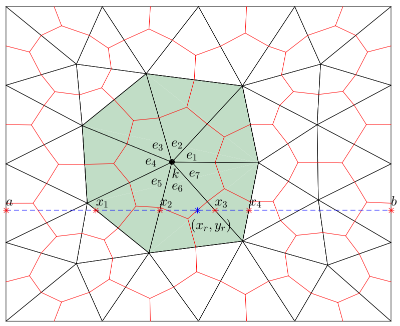

Figure 6: The illustration of line $y = y_r$ intersecting n_q points with the support domain Ω_{e_k} of $l_k(x, y)$, where (x_r, y_r) locates in Ω_{e_k}

$$l_k(x, y_r) = \begin{cases} 0, & a \le x \le x_1, \\ \varphi_{k5}(x, y_r), & x_1 \le x \le x_2, \\ \varphi_{k6}(x, y_r), & x_2 \le x \le x_3, \\ \varphi_{k7}(x, y_r), & x_3 \le x \le x_4, \\ 0, & x_4 \le x \le b. \end{cases}$$

Then

$$\begin{aligned} \frac{\partial^\alpha l_k(x, y_r)}{\partial x^\alpha}\bigg|_{x=x_r} &= \left(\frac{1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\int_a^x (x-\xi)^{-\alpha} l_k(\xi, y_r) d\xi\right)\bigg|_{x=x_r} \\ &= \left[\frac{1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\left(\int_a^{x_1} + \int_{x_1}^{x_2} + \int_{x_2}^{x}\right)(x-\xi)^{-\alpha} l_k(\xi, y_r) d\xi\right]\bigg|_{x=x_r} \\ &= \left[\frac{1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\left(\int_{x_1}^{x_2} + \int_{x_2}^{x}\right)(x-\xi)^{-\alpha} l_k(\xi, y_r) d\xi\right]\bigg|_{x=x_r}, \end{aligned} \tag{18}$$

and

$$\begin{aligned} \frac{\partial^\alpha l_k(x, y_r)}{\partial (-x)^\alpha}\bigg|_{x=x_r} &= \left(\frac{-1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\int_x^b (\xi-x)^{-\alpha} l_k(\xi, y_r) d\xi\right)\bigg|_{x=x_r} \\ &= \left[\frac{-1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\left(\int_x^{x_3} + \int_{x_3}^{x_4} + \int_{x_4}^{b}\right)(\xi-x)^{-\alpha} l_k(\xi, y_r) d\xi\right]\bigg|_{x=x_r} \\ &= \left[\frac{-1}{\Gamma(1-\alpha)}\frac{\partial}{\partial x}\left(\int_x^{x_3} + \int_{x_3}^{x_4}\right)(\xi-x)^{-\alpha} l_k(\xi, y_r) d\xi\right]\bigg|_{x=x_r}. \end{aligned} \tag{19}$$

If line $y = y_r$ intersects zero points with the support domain Ω_{e_k}, then we have

$$\frac{\partial^\alpha l_k(x, y_r)}{\partial x^\alpha}\bigg|_{x=x_r} = 0, \quad \frac{\partial^\alpha l_k(x, y_r)}{\partial (-x)^\alpha}\bigg|_{x=x_r} = 0. \tag{20}$$

The calculation of $\frac{\partial^\beta l_k(x,y)}{\partial y^\beta}$ and $\frac{\partial^\beta l_k(x,y)}{\partial (-y)^\beta}$ at (x_r, y_r) can be derived in a similar manner for the y direction. Finally, we summarise the whole computation process in the following algorithm (see Algorithm 1).

Algorithm 1 Unstructured mesh FVM for solving 2D SFDE-VC

Partition the convex domain Ω with unstructured triangular elements e_p and save the element information (node number, coordinates and element number);
for $p = 1, 2, \cdots, N_e$ **do**
    Find the barycenters of each triangular element e_p, form the control faces, sub-control volumes and save the sub-control volume information (the midpoint coordinates of each side of the triangular elements e_p, the midpoint coordinates (x_r, y_r) of each control faces, etc.);
    Calculate the areas of the sub-control volumes and control volumes, form matrix $\mathbf{A}$;
    for $k = 1, 2, \cdots, N_p$ **do**
        Find the support domain Ω_{e_k};
        Find the points of intersection by $y = y_r$ with Ω_{e_k} and calculate $\frac{\partial^\alpha l_k(x,y)}{\partial x^\alpha}\big|_{(x_r,y_r)}, \frac{\partial^\alpha l_k(x,y)}{\partial (-x)^\alpha}\big|_{(x_r,y_r)}$;
        Find the points of intersection by $x = x_r$ with Ω_{e_r} and calculate $\frac{\partial^\beta l_k(x,y)}{\partial y^\beta}\big|_{(x_r,y_r)}, \frac{\partial^\beta l_k(x,y)}{\partial (-y)^\beta}\big|_{(x_r,y_r)}$;
    end for
    Form the matrix $\mathbf{M}$;
    Form the vector $\mathbf{F}^n$;
end for
Solve the linear system (15) and obtain $\mathbf{U}^n$.

Remark 2.1. *When the boundary of the solution domain is nonconstant or curved, all of the above calculation is applicable as well.*

Here, we discuss the structure of matrix $\mathbf{M}$. Firstly, the matrix $\mathbf{M}$ generated by scheme (13) is sparse and not regular (see Figure 7). Then we explore the sparsity of matrix $\mathbf{M}$ for different h. Table 1 shows the size and density (nonzero entries percentage) of matrix $\mathbf{M}$ for different h where we can observe that as h decreases the density of matrix $\mathbf{M}$ reduces significantly. We can infer that when h is small enough, matrix $\mathbf{M}$ is extremely sparse and this facilitates the use of a sparse matrix storage format to reduce the memory usage of our computational method. Furthermore, we employ an efficient sparse iterative solver Bi-CGSTAB [59] to solve the linear system (15) (see Algorithm 2), which is more efficient than using the Gaussian elimination method. The CPU time comparison of the two methods is studied numerically in Example 3.1.

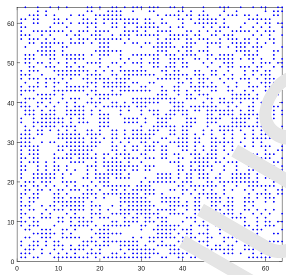

Figure 7: Sparsity pattern of matrix $\mathbf{M}$ for $h = 1.6759 \times 10^{-1}$. The size of $\mathbf{M}$ is 64×64. Blue points indicate the nonzero entries.

Table 1: The size and density of matrix $\mathbf{M}$ for different h on a square domain $[0,1] \times [0,1]$

h	Size	Density
5.26[illegible]E-01	4×4	100%
3.1[illegible]23E-01	15×15	86.667%
1.675[illegible]E-01	64×64	57.715%
[illegible].66[illegible]2E-02	258×258	34.002%
4.3[illegible]19E-02	1115×1115	17.705%
2.3[illegible]E-02	5255×5255	8.517%

3. Discussion of Numerical Results

In this section, we provide some numerical examples to verify the effectiveness of our method presented in section 2. We adopt linear polynomials on triangles and define h as the maximum length of the triangle edges. N_e is taken as the number of triangles in $\mathcal{T}_h$. Here, the numerical computations were carried out using MATLAB R2014b on a Dell desktop with configuration: Intel(R) Core(TM) i7-4790, 3.60 GHz and 16.0 GB RAM. We use the following formula to calculate the convergence order:

$$\text{Order} = \frac{\log(E(h_1)/E(h_2))}{\log(h_1/h_2)},$$

where E is the L_2 or L_∞ error.

Example 3.1. *Firstly, we consider the following 2D SFDE-VC on a rectangular domain*

$$\begin{aligned}\frac{\partial u(x,y,t)}{\partial t} =& \frac{\partial}{\partial x}\left[K_1(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial x^\alpha} - K_2(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial(-x)^\alpha}\right] \\ &+ \frac{\partial}{\partial y}\left[K_3(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial y^\beta} - K_4(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial(-y)^\beta}\right] \\ &+ f(x,y,t), \quad (x,y,t) \in \Omega \times (0,T],\end{aligned}$$

Algorithm 2 The Bi-CGSTAB algorithm

```
Define A_0 = A − τM, use a sparse matrix storage format to store A_0;
In each time level t_n, x_0 = U^{n−1}, b = AU^{n−1} + τAF^n;
Compute r_0 = b − A_0 x_0, r̂_0 is an arbitrary vector, such that (r̂_0, r_0) ≠ 0. We choose r̂_0 = r_0;
Let ρ_0 = α_0 = ω_0 = 1, v_0 = p_0 = 0;
for i = 1, 2, 3, ··· , do
    ρ_i = (r̂_0, r_{i−1});
    β_0 = (ρ_i/ρ_{i−1})(α_{i−1}/ω_{i−1});
    p_i = r_{i−1} + β_0(p_{i−1} − ω_{i−1}v_{i−1});
    v_i = A_0 p_i, α_i = ρ_i/(r̂_0, v_i);
    s = r_{i−1} − α_i v_i, t_0 = A_0 s;
    ω_i = (t_0, s)/(t_0, t_0);
    x_i = x_{i−1} + α_i p_i + ω_i s;
    if x_i is accurate enough then quit;
    r_i = s − ω_i t_0;
end for
U^n = x_i.
```

subject to

$$
\begin{aligned}
&u(x,y,0) = x^2(1-x)^2y^2(1-y)^2, \quad (x,y)\in\overline{\Omega},\\
&u(x,y,t) = 0, \quad (x,y,t)\in\partial\Omega\times[0,T].
\end{aligned}
$$

where $\Omega=(0,1)\times(0,1)$*,* $T=1$*,*

$$
\begin{aligned}
f(x,y,t) &= 2tx^2(1-x)^2y^2(1-y)^2 - \Big[\frac{\partial K_1(x,y,t)}{\partial x}\cdot p(x,\alpha) + K_1(x,y,t)\cdot p(x,1+\alpha)\\
&\quad - \frac{\partial K_2(x,y,t)}{\partial x}\cdot p(1-x,\alpha) + K_2(x,y,t)\cdot p(1-x,1+\alpha)\Big]y^2(1-y)^2(t^2+1)\\
&\quad - \Big[\frac{\partial K_3(x,y,t)}{\partial y}\cdot p(y,\beta) + K_3(x,y,t)\cdot p(y,1+\beta) - \frac{\partial K_4(x,y,t)}{\partial y}\cdot p(1-y,\beta)\\
&\quad + K_4(x,y,t)\cdot p(1-y,1+\beta)\Big]x^2(1-x)^2(t^2+1),\\
p(z,r) &= \frac{\Gamma(3)}{\Gamma(3-r)}z^{2-r} - \frac{2\Gamma(4)}{\Gamma(4-r)}z^{3-r} + \frac{\Gamma(5)}{\Gamma(5-r)}z^{4-r}.
\end{aligned}
$$

This is a two-dimensional anomalous diffusion model, which can describe anomalous transport in heterogeneous porous media and can be used to explain the region-scale anomalous dispersion with heavy tails [28].

The exact solution of this problem is given by $u(x,y,t)=(t^2+1)x^2(1-x)^2y^2(1-y)^2$. Figure 8 shows the rectangular domain partitioned by unstructured triangular meshes and control volumes for different h. Here, we consider three different coefficient cases [30]: linear coefficients $K_1(x,y,t)=2-x$, $K_2(x,y,t)=2+x$, $K_3(x,y,t)=2-y$, $K_4(x,y,t)=2+y$, quadratic coefficients $K_1(x,y,t)=2-x^2$, $K_2(x,y,t)=2+x^2$, $K_3(x,y,t)=2-y^2$, $K_4(x,y,t)=2+y^2$ and exponential coefficients $K_1(x,y,t)=3-e^x$, $K_2(x,y,t)=3+e^x$, $K_3(x,y,t)=3-e^y$, $K_4(x,y,t)=3+e^y$. The numerical results are given in Tables 2 to 4. Table 2 illustrates the L_2 error, L_∞ error and corresponding convergence order of h for the linear coefficient case for different α, β with $\tau=10^{-3}$ at $t=1$. Tables 3 and 4 show the L_2 error, L_∞ error and corresponding convergence order of h for the quadratic coefficient case and exponential coefficient case, respectively. From these tables we can see that the convergence order of both the L_2 error and L_∞ error is $2-\max\{\alpha,\beta\}$ order [27] and the numerical results are in excellent agreement with the exact solution, which demonstrates the effectiveness of the numerical method. We can also observe that with h deceasing, the CPU time grows considerably, which we believe is mainly due to the non-locality of the fractional derivative of the basis function and the computational cost to generate the matrix $\mathbf{M}$. In addition, we give a comparison between the Bi-CGSTAB and Gaussian elimination. In the Bi-CGSTAB solver, we set 10^{-10} as the stopping criterion and the maximum iteration number is 10^2. Table 5 displays the consumed CPU time of these two algorithms at $t = 1$ with $\tau = 10^{-3}$, $\alpha = 0.3$, $\beta = 0.5$, $K_1(x, y, t) = 2 - x$, $K_2(x, y, t) = 2 + x$, $K_3(x, y, t) = 2 - y$, $K_4(x, y, t) = 2 + y$ for different h. Compared to Gaussian elimination, Bi-CGSTAB has significantly reduced 90% of the computational time for $h = 4.3719 \times 10^{-2}$. Another advantage of Bi-CGSTAB to be mentioned is that the average iteration number does not appear to increase significantly as h decreases. Here, the average iteration number is approximately 10 regardless of the model dimensions. We conclude that the Bi-CGSTAB solver is more efficient than Gaussian elimination for solving this problem.

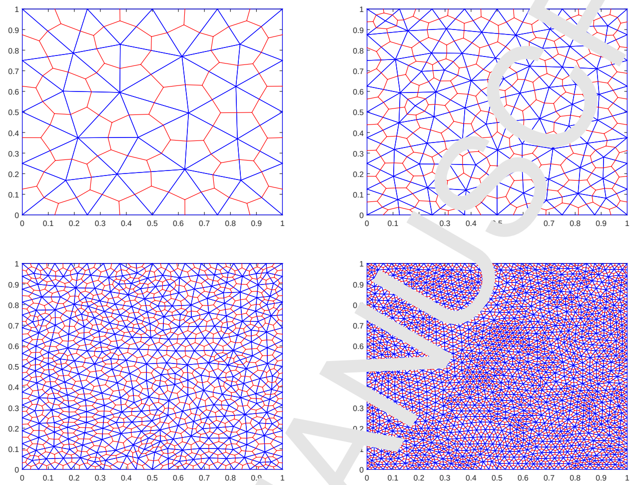

Figure 8: The rectangular domain partitioned by unstructured meshes with control volumes for $h \approx 3.1123 \times 10^{-1}, 1.6759 \times 10^{-1}, 8.6682 \times 10^{-2}, 4.3719 \times 10^{-2}$, respectively

Table 2: The L_2 error, L_∞ error, convergence order and CPU time of h with $\tau = 10^{-3}$ for the linear coefficient case at $t = 1$

	h	L_2 error	Order	L_∞ error	Order	Time
	3.1123E-01	3.5684E-04	–	1.4774E-03	–	4.90s
$\alpha = 0.3$	1.6759E-01	1.0880E-04	1.92	4.3735E-04	1.97	19.50s
$\beta = 0.5$	8.6682E-02	2.2391E-05	2.40	1.3895E-04	1.74	2.30min
	4.3719E-02	6.9379E-06	1.71	3.7632E-05	1.91	28.42min
	3.1123E-01	3.7935E-04	–	1.4827E-03	–	4.91s
$\alpha = 0.4$	1.6759E-01	1.2435E-04	1.80	4.2971E-04	2.00	19.98s
$\beta = 0.8$	8.6682E-02	2.5152E-05	2.42	1.3725E-04	1.73	2.36min
	4.3719E-02	7.2675E-06	1.81	3.5722E-05	1.97	28.56min
	3.1123E-01	3.9259E-04	–	1.3844E-03	–	4.91s
$\alpha = 0.7$	1.6759E-01	1.4100E-04	1.65	4.1957E-04	1.93	19.87s
$\beta = 0.8$	8.6682E-02	2.8670E-05	2.42	1.4117E-04	1.65	2.37min
	4.3719E-02	7.5385E-06	1.95	3.3666E-05	2.09	28.47min

Example 3.2. *Next, we consider the following two-dimensional Riesz space fractional diffusion equation on a circular domain, which can be used to describe the propagation of the electrical potential in heterogeneous cardiac tissue [49, 52, 58].*

$$\begin{cases} \frac{\partial u(x,y,t)}{\partial t} = K_x \frac{\partial^{1+\alpha} u(x,y,t)}{\partial |x|^{1+\alpha}} + K_y \frac{\partial^{1+\beta} u(x,y,t)}{\partial |y|^{1+\beta}} + f(x, y, t), \quad (x, y, t) \in \Omega \times (0, T], \\ u(x, y, 0) = (x^2 + y^2 - 1)^2, \quad (x, y) \in \overline{\Omega}, \\ u(x, y, t) = 0, \quad (x, y, t) \in \partial\Omega \times [0, T], \end{cases} \tag{21}$$

Table 3: The L_2 error, L_∞ error, convergence order and CPU time of h with $\tau = 10^{-3}$ for the quadratic coefficient case at $t = 1$

	h	L_2 error	Order	L_∞ error	Order	Time
	3.1123E-01	3.1608E-04	–	1.3430E-03	–	4.97s
$\alpha = 0.3$	1.6759E-01	1.0064E-04	1.85	4.0906E-04	1.92	20.48s
$\beta = 0.5$	8.6682E-02	2.0661E-05	2.40	1.3852E-04	[illegible]	2.45min
	4.3719E-02	6.2709E-06	1.74	3.7584E-05	1.91	28.69min
	3.1123E-01	3.6299E-04	–	1.4108E-03	–	4.88s
$\alpha = 0.4$	1.6759E-01	1.2145E-04	1.77	4.1614E-04	1.97	20.51s
$\beta = 0.8$	8.6682E-02	2.4646E-05	2.42	1.3823E-04	1.67	2.46min
	4.3719E-02	6.7517E-06	1.89	3.3858E-05	2.06	28.78min
	3.1123E-01	3.8524E-04	–	[illegible]	–	4.97s
$\alpha = 0.7$	1.6759E-01	1.3952E-04	1.64	[illegible]	1.93	20.56s
$\beta = 0.9$	8.6682E-02	2.8522E-05	2.41	[illegible]	1.60	2.44min
	4.3719E-02	7.1520E-06	2.02	3.1880E-05	2.17	28.68min

Table 4: The L_2 error, L_∞ error, convergence order and CPU time of h with $\tau = 10^{-3}$ for the exponential coefficient case at $t = 1$

	h	L_2 error	Order	L_∞ error	Order	Time
	3.1123E-01	5.1809E-04	[illegible]	1.9033E-03	–	4.97s
$\alpha = 0.3$	1.6759E-01	1.6296E-04	1.87	5.3973E-04	2.04	20.62s
$\beta = 0.5$	8.6682E-02	3.8817E-05	2.18	1.6032E-04	1.84	2.45min
	4.3719E-02	1.1574E-05	1.77	4.8226E-05	1.76	28.46min
	3.1123E-01	4.5022E-04	–	1.6750E-03	–	4.93s
$\alpha = 0.4$	1.6759E-01	[illegible]	1.79	1.0117E-04	2.01	20.52s
$\beta = 0.8$	8.6682E-02	[illegible]	2.24	4.8309E-04	1.84	2.45min
	4.3719E-02	[illegible]	1.62	4.3016E-05	1.76	28.66min
	3.1123E-01	4.2412E-04	–	1.4994E-03	–	4.93s
$\alpha = 0.7$	1.6759E-01	[illegible]	1.65	4.6520E-04	1.89	20.50s
$\beta = 0.9$	8.6682E-02	[illegible]	2.31	1.4533E-04	1.76	2.45min
	4.3719E-02	1.0565E-05	1.68	4.0322E-05	1.87	28.56min

where $\Omega = \{(x,y)|x^2+y^2<1\}$, $K_x = 1$, $K_y = 1$, $T = 1$,

$$
\begin{aligned}
f(x,y,t) &= -e^{-t}(x^2+y^2-1)^2 \\
&+ \frac{e^{-t}}{2\cos((1+\alpha)/2\pi)}\Big[\Big(f_1(x,a_0,\alpha)+g_1(x,b_0,\alpha)\Big)+(2y^2-2)\Big(f_2(x,a_0,\alpha)+g_2(x,b_0,\alpha)\Big) \\
&+ (y^2-1)^2\Big(f_3(x,a_0,\alpha)+g_3(x,b_0,\alpha)\Big)\Big] \\
&- \frac{e^{-t}}{2\cos((1+\beta)/2\pi)}\Big[\Big(f_1(y,c_0,\beta)+g_1(y,d_0,\beta)\Big)+(2x^2-2)\Big(f_2(y,c_0,\beta)+g_2(y,d_0,\beta)\Big) \\
&+ (x^2-1)^2\Big(f_3(y,c_0,\beta)+g_3(y,d_0,\beta)\Big)\Big],
\end{aligned}
$$

$$
\begin{aligned}
&a_0 = -\sqrt{1-y^2},\ b_0 = \sqrt{1-y^2},\ c_0 = -\sqrt{1-x^2},\ d_0 = \sqrt{1-x^2}, \\
&f_1(x,a,\alpha) = {}_aD_x^{1+\alpha}(x^4),\ f_2(x,a,\alpha) = {}_aD_x^{1+\alpha}(x^2),\ f_3(x,a,\alpha) = {}_aD_x^{1+\alpha}(1), \\
&g_1(x,b,\alpha) = {}_xD_b^{1+\alpha}(x^4),\ g_2(x,b,\alpha) = {}_xD_b^{1+\alpha}(x^2),\ g_3(x,b,\alpha) = {}_xD_b^{1+\alpha}(1).
\end{aligned}
$$

The exact solution is given by $u(x,y,t) = e^{-t}(x^2+y^2-1)^2$. Figure 9 shows the circular domain partitioned by

Table 5: Comparison of the consumed CPU time of Gaussian elimination versus Bi-CGSTAB

N_e	h	Gauss elimination	Bi-CGSTAB
44	3.1123E-01	4.90s	4.90s
158	1.6759E-01	22.57s	19.50s
578	8.6682E-02	5.39min	2.30min
2356	4.3719E-02	5.48h	28.42min

unstructured triangular meshes and control volumes for different h. In [52], Yang et al. applied the Galerkin finite element method for solving the two-dimensional Riesz space fractional diffusion equation with a nonlinear source term on convex domains. They developed an algorithm to form the stiffness matrix on triangular meshes, which can deal with space fractional derivatives on any convex domain. Here, we will make a comparison between our method (CVM) and Yang's method (FEM) for solving the two-dimensional Riesz space fractional diffusion equation (21) on a circular domain using the same triangular meshes. Firstly, we present a comparison of the density of the two stiffness matrices generated by FEM and CVM for different h in Table 6. We can see that with h decreasing the density of the two stiffness matrices reduces significantly. Compared to the stiffness matrix generated by FEM, the stiffness matrix generated by CVM is slightly more sparse. Next, we present a comparison of the error and convergence. Table 7 displays the L_2 error, L_∞ error and corresponding convergence order of h for different α, β with $\tau = 10^{-3}$ at $t = 1$ by applying FEM. Table 8 highlights the error and convergence order by using FVM. We can see that the accuracy of our method is similar to FEM, both of which are second order. Then, we present a comparison of CPU time for the two methods in Table 9 both using the Bi-CGSTAB solver. We choose $\alpha = \beta = 0.8$ and $\tau = 10^{-3}$ at $t = 1$ to observe the running time for different h. We observe that compared to the running time of FEM, CVM can reduce the running time significantly, which illustrates that CVM is more effective for solving the two-dimensional Riesz space fractional diffusion equation on convex domains. This is mainly due to the bilinear form in [52] that involves 8 fractional derivative terms and the approximation of two-fold multiple integrals, which are approximated by Gauss quadrature, while for CVM we only need to calculate 4 fractional derivative terms and the approximation of line integrals. In addition, we give a comparison of the exact solution $u(x, y, t)$ and numerical solution $u_h(x, y, t)$ in Figure 10 and the error plot of $u(x, y, t) - u_h(x, y, t)$ in Figure 11 for $h = 4.5873 \times 10^{-2}$, $\alpha = \beta = 0.8$ with $\tau = 10^{-3}$ at $t = 1$ by applying CVM. We can see that the numerical solution is in excellent agreement with the exact solution, which demonstrates the effectiveness of our numerical method again.

Table 6: The comparison of the density of stiffness matrix generated by FEM and CVM for different h

N_e	h	Size	FEM	CVM
174	2.8917E-01	74×74	65.413 %	55.332%
57[illegible]	1.6444E-01	260×260	41.814 %	33.521%
2310	8.6550E-02	1104×1104	22.233 %	17.469%
8[illegible]44	4.5873E-02	4271×4271	11.712 %	9.107%

Example 3.3. *Finally, we consider the following 2D SFDE-VC without a source term on different convex domains*

$$\frac{\partial u(x,y,t)}{\partial t} = \frac{\partial}{\partial x}\left[K_1(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial x^\alpha} - K_2(x,y,t)\frac{\partial^\alpha u(x,y,t)}{\partial(-x)^\alpha}\right]$$
$$+\frac{\partial}{\partial y}\left[K_3(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial y^\beta} - K_4(x,y,t)\frac{\partial^\beta u(x,y,t)}{\partial(-y)^\beta}\right], \quad (x,y,t) \in \Omega \times (0,T],$$

subject to

$$u(x,y,0) = 100, \quad (x,y) \in \Omega,$$
$$u(x,y,t) = 0, \quad (x,y,t) \in \partial\Omega \times [0,T].$$

where $K_1(x,y,t) = 2 - x$, $K_2(x,y,t) = 2 + x$, $K_3(x,y,t) = 2 - y$, $K_4(x,y,t) = 2 + y$, $T = 0.5$.

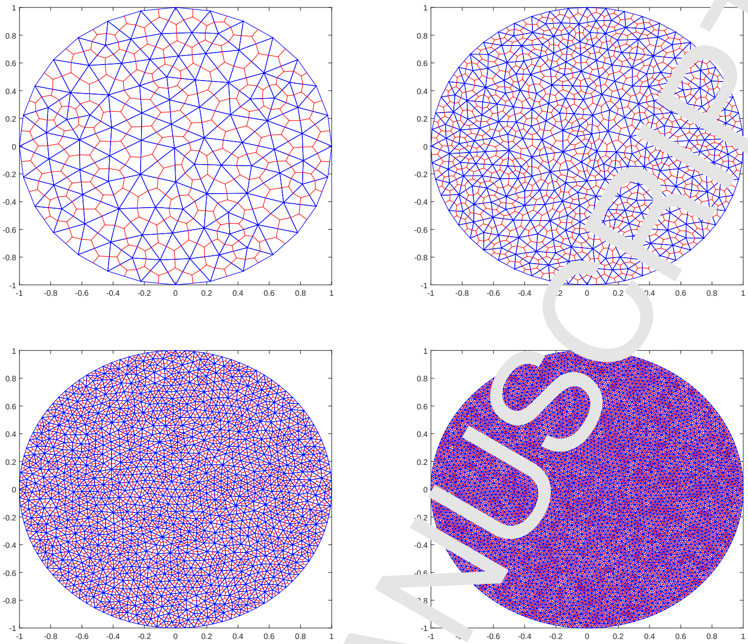

Figure 9: The unstructured meshes with control volumes for
$h \approx 2.8917 \times 10^{-1}, 1.6444 \times 10^{-1}, 8.65[illegible]0 \times 10^{-2}, 4.5873 \times 10^{-2}$, respectively

Here, we choose $\alpha = \beta = 0.8$ and $\tau = 10^{-3}$ to observe the diffusion behavior of $u(x, y, t)$. Figure 12 shows the different diffusion profiles of $u(x, y, t)$ at $t = 0.5$ on different convex domains. We can see that the diffusive behaviour of $u(x, y, t)$ is different on different convex domains, in which the diffusive velocity on domain 1 is the fastest and the diffusive velocity on domain 4 is the slowest. We also can observe that our method is effective and is applicable for all these convex domains.

4. Conclusions

In this paper, we considered the unstructured mesh control volume method for the two-dimensional space fractional diffusion equation with variable coefficients on convex domains. We partitioned the irregular convex domain using triangular meshes. Then we constructed the control volumes and solved the space fractional diffusion equation by utilising the finite volume method. Finally, numerical examples on irregular convex domains were studied, which verified the effectiveness and reliability of the method. We concluded that the numerical method can be extended to other arbitrarily shaped convex domains. Furthermore, according to the property of the stiffness matrix generated by the finite volume method, we chose a suitable sparse matrix format for the stiffness matrix and utilised the Bi-CGSTAB iterative method to solve the linear system, which is more efficient than using the Gauss elimination method. In addition, we made a comparison of our method with the finite element method proposed in [52], which demonstrated that our method can reduce CPU time significantly while retaining the same accuracy and approximation property as the finite element method. In future work, we shall investigate the unstructured mesh control volume method applied to other fractional problems on irregular convex domains, such as the two-dimensional multi-term time-space fractional diffusion equation with variable coefficients, or three-dimensional space fractional diffusion equations with variable coefficients.

Acknowledgment

Authors Liu and Turner wish to acknowledge that this research was partially supported by the Australian Research Council (ARC) via the Discovery Projects (DP180103858 and DP190101889), and Natural Natural Science

Table 7: The L_2 error, L_∞ error and convergence order of h for FEM with $\tau = 10^{-3}$ at $t = 1$

FEM	h	L_2 error	Order	L_∞ error	Order
	2.8917E-01	6.7022E-03	–	5.8841E-03	–
$\alpha = 0.80$	1.6444E-01	2.0787E-03	2.07	2.8557E-03	1.28
$\beta = 0.80$	8.6550E-02	5.2077E-04	2.16	8.1791E-04	1.95
	4.5873E-02	1.3554E-04	2.12	2.3520E-04	1.96
	2.8917E-01	6.9018E-03	–	5.5925E-03	–
$\alpha = 0.70$	1.6444E-01	2.1713E-03	2.05	2.7718E-03	1.24
$\beta = 0.90$	8.6550E-02	5.4452E-04	2.16	7.9048E-04	1.95
	4.5873E-02	1.4147E-04	2.12	2.2242E-04	2.00

Table 8: The L_2 error, L_∞ error and convergence order of h for CVM with $\tau = 10^{-3}$ at $t = 1$

CVM	h	L_2 error	Order	L_∞ error	Order
	2.8917E-01	1.4782E-02	–	2.1786E-02	–
$\alpha = 0.80$	1.6444E-01	4.5014E-03	2.11	7.5230E-03	1.88
$\beta = 0.80$	8.6550E-02	1.2275E-03	2.02	1.8279E-03	2.20
	4.5873E-02	3.4069E-04	2.02	5.4557E-04	1.90
	2.8917E-01	1.4950E-02	–	2.1864E-02	–
$\alpha = 0.70$	1.6444E-01	4.5530E-03	2.11	7.6462E-03	1.86
$\beta = 0.90$	8.6550E-02	1.2566E-03	2.01	1.8659E-03	2.20
	4.5873E-02	3.4893E-04	2.02	5.4606E-04	1.94

Foundation of China (Grant No.11772046). Author Feng wishes to acknowledge that this research was partially supported by the National Natural Science Foundation of China (Grant No.11801060). Author Turner wishes to acknowledge that this research was also partially supported by the Australian Research Council (ARC) via the Discovery Project DP150103675.

References

[1] I.M. Sokolov, J. Klafter, A. Blumen, Fractional kinetics, Phys. Today Nov. (2002) 28-53.

[2] R.L. Magin, Fractional calculus in bioengineering, Begell House Publisher., Inc., Connecticut, 2006.

[3] F. Liu, P. Zhuang, Q. Liu, Numerical Methods of Fractional Partial Differential Equations and Applications, Science Press, China, November 2015 ISBN 978-7-03-046335-7.

[4] E. Scalas, R. Gorenflo, F. Mainardi, Fractional calculus and continuous-time finance, Phys. A: Stat. Mech. Appl. 284 (2000) 376-384.

[5] L. Feng, F. Liu, I. Turner, P. Zhuang, Numerical methods and analysis for simulating the flow of a generalized Oldroyd-B fluid between two infinite parallel rigid plates, International Journal of Heat and Mass Transfer 115 (2017) 1309-1320.

[6] A. A. Kilbas, H.M. Srivastava, J.J. Trujillo, Theory and applications of fractional differential equations, Elsevier, North-Holland, 2006.

[7] K. Diethelm, The Analysis of Fractional Differential Equations: An Application-Oriented Exposition Using Differential Opereators of Caputo Type, Springer-Verlag, Berlin, 2010.

[8] H. Zhang, X. Jiang, X. Yang, A time-space spectral method for the time-space fractional Fokker-Planck equation and its inverse problem, Appl. Math. Comput. 320 (2018) 302-318.

Table 9: The comparison of running time between FEM and CVM for different h with $\alpha = \beta = 0.80$, $\tau = 10^{-3}$ at $t = 1$

N_e	h	FEM	CVM
174	2.8917E-01	3.49 min	35.01 s
570	1.6444E-01	12.90 min	2.63min
2310	8.6550E-02	1.38 h	28.41min
8744	4.5873E-02	17.89h	6.59h

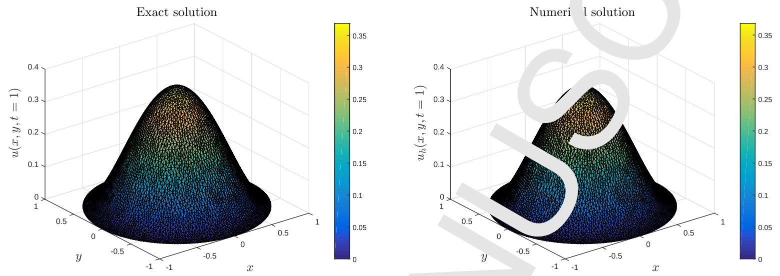

Figure 10: The comparison of the exact solution $u(x, y, t)$ and numerical solution $u_h(x, y, t)$ for $h = 4.5873 \times 10^{-2}$, $\alpha = \beta = 0.8$ with $\tau = 10^{-3}$ at $t = 1$

[9] H. Zhang, X. Jiang, C. Wang, W Fan, Galerkin-Legendre spectral schemes for nonlinear space fractional Schrödinger equation, Numerical Algorithms 79 (2018) 337-356.

[10] M. Zhang, M. Shen, F. Liu, H. Zhang, A new time and spatial fractional heat conduction model for Maxwell nanofluid in porous medium, Computers and Mathematics with Applications,(accepted on 10 Jan., 2019), in press, https://doi.org/10.1016/j.camwa.2019.01.006.

[11] H. Zhang, F. Liu, I. Turner, S. Chen, Q. Yang, Numerical simulation of a Finite Moment Log Stable model for a European call option. Numerical Algorithms, 75(3) (2017) 569-585.

[12] M. Zheng, F. Liu, Q. Liu, K. Burrage, M.J. Simpson, A Novel Numerical Approximation of the Time Fractional Reaction-Diffusion Equation with a Moving Boundary, Journal of Computational Physics, 338, (2017), 493-510.

[13] M. M. Meerschaert, C. Tadjeran, Finite difference approximations for fractional advection-dispersion flow equations, J. Comput. Appl. Math. 172 (2004) 65-77.

[14] F. Liu, V. Anh, I. Turner, Numerical solution of the space fractional Fokker-Planck Equation, Journal of Computational and Applied Mathematics 166 (2004) 209-219.

[15] E. Barkai, R. Metzler, J. Klafter, From continuous time random walks to the fractional Fokker-Planck equation, Phys. Rev. E 61 (2000) 132-138.

[16] A.V. Chechkin, J. Klafter, I.M. Sokolov, Fractional Fokker-Planck equation for ultraslow kinetics, Europhys. Lett. 63 (2003) 326-332.

[17] M. M. Meerschaert, C. Tadjeran, Finite difference approximations for two-sided space-fractional partial differential equations, Appl. Numer. Math. 56 (2006) 80-90.

[18] Y. Zhang, D. A. Benson, M. M. Meerschaert, E.M. LaBolle, Space-fractional advection-dispersion equations with variable parameters: Diverse formulas, numerical solutions, and application to the Macrodispersion Experiment site data, Water Resour. Res. 43 (2007) W05439, doi:10.1029/2006WR004912.

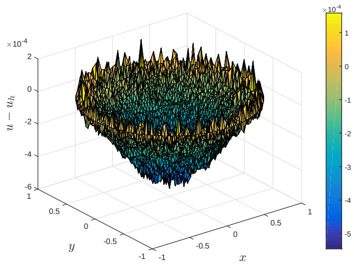

Figure 11: The error plot of $u(x, y, t) - u_h(x, y, t)$ for $h = 4.5873 \times 10^{-2}$, $\alpha = \beta = 0.8$ with $\tau = 10^{-3}$ at $t = 1$

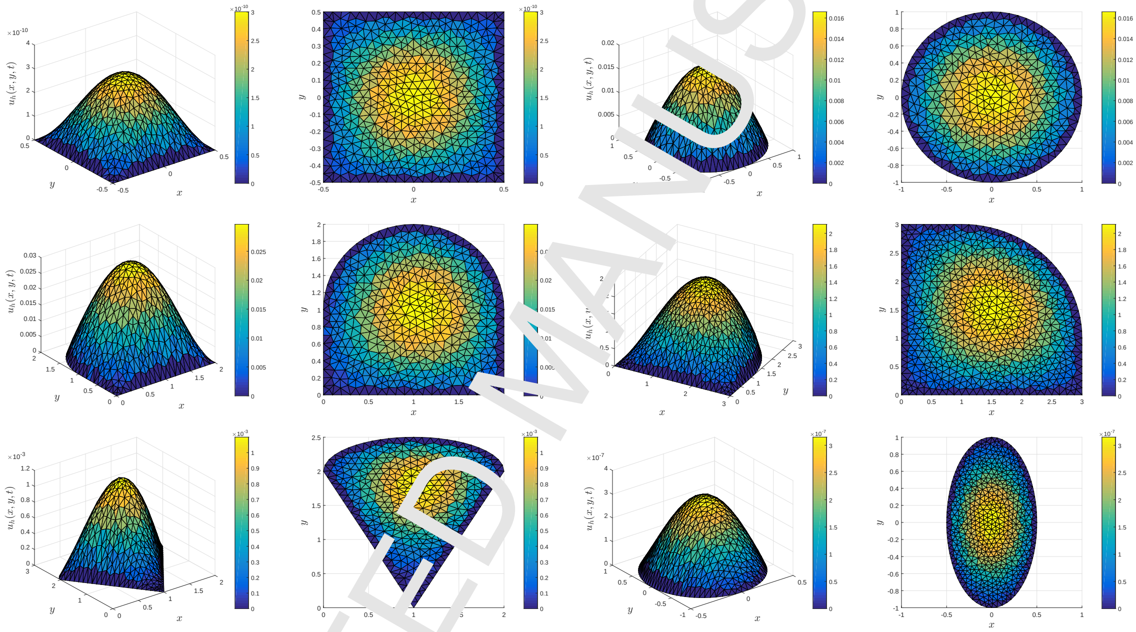

Figure 12: The diffusion profiles of $u(x, y, t)$ at $t = 0.5$ on different convex domains with $\alpha = \beta = 0.8$ and $\tau = 10^{-3}$

[19] Z. Ding, A. Xiao, M. Li, Weighted finite difference methods for a class of space fractional partial differential equations with variable coefficients, J. Comput. Appl. Math. 233 (2010) 1905-1914.

[20] T. Moroney, Q. Yang, A banded preconditioner for the two-sided, nonlinear space-fractional diffusion equation, Comput. Math. Appl. 66 (2013) 659-667.

[21] T. Moroney, Q. Yang, Efficient solution of two-sided nonlinear space-fractional diffusion equations using fast Poisson preconditioners, J. Comput. Phys. 246 (2013) 304-317.

[22] M. Chen, W. Deng, A second-order numerical method for two-dimensional two-sided space fractional convection diffusion equation, Appl. Math. Model. 38 (2014) 3244-3259.

[23] H. Wang, X. Zhang, A high-accuracy preserving spectral Galerkin method for the Dirichlet boundary-value problem of variable-coefficient conservative fractional diffusion equations, J. Comput. Phys. 281 (2015) 67-81.

[24] F. Liu, P. Zhuang, I. Turner, K. Burrage, V. Anh, A new fractional finite volume method for solving the fractional diffusion equation, Applied Mathematical Modelling, 38, (2014), 3871-3878.

[25] J. Li, F. Liu, L. Feng, I. Turner, A novel finite volume method for the Riesz space distributed-order advection-diffusion equation, Applied Mathematical Modelling, 46, (2017), 536-553.

[26] J. Li, F. Liu, L. Feng, I. Turner, A novel finite volume method for the Riesz space distributed-order diffusion equation, Computers and Mathematics with Applications, 74(4), (2017), 772-783.

[27] L. Feng, P. Zhuang, F. Liu and I. Turner, Stability and convergence of a new finite volume method for a two-sided space-fractional diffusion equation, Appl. Math. Comput. 257 (2015) 52-65.

[28] S. Chen, F. Liu, X. Jiang, I. Turner, K. Burrage, Fast finite difference approximation for identifying parameters in a two-dimensional space-fractional nonlocal model with variable diffusivity coefficients, SIAM J. Numer. Anal. 54 (2016) 606-624.

[29] J. Jia, H. Wang, A fast finite volume method for conservative space-fractional diffusion equations in convex domains, J. Comput. Phys. 310 (2016) 63-84.

[30] L. Feng, P. Zhuang, F. Liu, I. Turner, V. Anh, J. Li, A fast second-order accurate method for a two-sided space-fractional diffusion equation with variable coefficients, Comput. Math. Appl. 73 (2017) 1155-1171.

[31] R. Chen, F. Liu, V. Anh, Numerical methods and analysis for a multi-term time-space variable-order fractional advection-diffusion equations and applications, Journal of Computational and Applied Mathematics, 352, (2019), 437-452.

[32] F. Liu, M.M. Meerschaert, R. McGough, P. Zhuang, Q. Liu, Numerical methods for solving the multi-term time fractional wave equations, Fractional Calculus and Applied Analysis, 16(1), (2013), 9-25.

[33] C. Li, Z. Zhao, Y.Q. Chen, Numerical approximation of nonlinear fractional differential equations with subdiffusion and superdiffusion, Comput. Math. Appl. 62 (2011) 855-875.

[34] F. Zeng, Z. Zhang, G.E. Karniadakis, Second-order numerical methods for multi-term fractional differential equations: smooth and non-smooth solutions, Comput. Methods Appl. Mech. Eng. 327 (2017) 478-502.

[35] F. Liu, P. Zhuang, V. Anh, I. Turner, K. Burrage, Stability and convergence of the difference methods for the space-time fractional advection-diffusion equation, Appl. Math. Comput. 191 (2007) 12-20.

[36] Y. Liu, Y. Du, H. Li, F. Liu, Y. Wang, Some second-order θ schemes combined with finite element method for nonlinear fractional Cable equation, Numerical Algorithm 80 (2019) 533-555.

[37] W. Bu, Y. Tang, J. Yang, Galerkin finite element method for two-dimensional Riesz space fractional diffusion equations, J. Comput. Phys. 276 (2014) 26-38.

[38] Y. Liu, Y. Du, H. Li, S. He, W. Gao, Finite difference/finite element method for a nonlinear time-fractional fourth-order reaction-diffusion problem, Comput. Math. Appl. 70(4) (2015) 573-591.

[39] L. Feng, P. Zhuang, F. Liu, I. Turner, Y. Gu, Finite element method for space-time fractional diffusion equation, Numerical Algorithm 72 (2016) 749-767.

[40] B. Jin , R. Lazarov , Z. Zhou , Error estimates for a semidiscrete finite element method for fractional order parabolic equations, SIAM J. Numer. Anal. 51 (2013) 445-466.

[41] M. Zheng, F. Liu, V. Anh, I. Turner, A high order spectral method for the multi-term time-fractional diffusion equations, Applied Mathematical Modelling 40 (2016) 4970-4985.

[42] N. Ford, J. Xiao, Y. Yan, A finite element method for time fractional partial differential equations, Fractional Calculus and Applied Analysis 14 (2011) 454-474.

[43] M. Zheng, F. Liu, I. Turner, V. Anh, A novel high order space-time spectral method for the time fractional Fokker-Planck equation, SIAM J. Sci. Computing 37 (2015) A701-A724.

[44] F. Liu, L. Feng, V. Anh, J. Li, Unstructured-mesh Galerkin finite element method for the two-dimensional multi-term time-space fractional Bloch-Torrey equations on irregular convex domains, Computers and Mathematics with Applications, (accepted on 10 Jan., 2019), https://doi.org/10.1016/j.camwa.2019.01.007.

[45] W. Fan, X. Jiang, F. Liu, V. Anh, The unstructured mesh finite element method for the two-dimensional multi-term time-space fractional diffusion-wave equation on an irregular convex domain, Journal of Scientific Computing, 77, (2018), 27-52.

[46] Q. Yang, I. Turner, T. Moroney, F. Liu, A finite volume scheme with preconditioned Lanczos method for two-dimensional space-fractional reaction-diffusion equations, Appl. Math. Model. 38 (2014) 3755-3762.

[47] K. Burrage, N. Hale, D. Kay, An efficient implicit FEM scheme for fractional-in-space reaction-diffusion equations, SIAM J. Sci. Comput. 34 (2012) A2145-A2172.

[48] L. Qiu, W. Deng, J.S. Hesthaven, Nodal discontinuous Galerkin methods for fractional diffusion equations on 2D domain with triangular meshes, J. Comput. Phys. 298 (2015) 678-694.

[49] F. Liu, P. Zhuang, I. Turner, V. Anh, K. Burrage, A semi-alternating direction method for a 2-D fractional FitzHugh-Nagumo monodomain model on an approximate irregular domain, J. Comput. Phys. 293 (2015) 252-263.

[50] S. Qin, F. Liu, I. Turner, Q. Yang, Q. Yu, Modelling anomalous diffusion using fractional Bloch-Torrey equations on approximate irregular domains, Comput. Math. Appl. 75 (2018) 7-21.

[51] S. Karaa, K. Mustapha, A.K. Pani, Finite volume element method for two-dimensional fractional subdiffusion problems, IMA J. Numer. Anal. 37 (2016) 945-964.

[52] Z. Yang, Z. Yuan, Y. Nie, J. Wang, X. Zhu, F. Liu, Finite element method for nonlinear Riesz space fractional diffusion equations on irregular domains, J. Comput. Phys. 330 (2017) 863-883.

[53] W. Fan, F. Liu, X. Jiang, I. Turner, A novel unstructured mesh finite element method for solving the time-space fractional wave equation on a two-dimensional irregular convex domain, Fract. Calc. Appl. Anal. 20 (2017) 352-383.

[54] L. Feng, F. Liu, I. Turner, Q. Yang, P. Zhuang, Unstructured mesh finite difference/finite element method for the 2D time-space Riesz fractional diffusion equation on irregular convex domains, Applied Mathematical Modelling 59 (2018) 441-463.

[55] K. N. Le, W. McLean, B. Lamichhane, Finite element approximation of a time-fractional diffusion problem for a domain with a re-entrant corner, The ANZIAM Journal 59 (2017) 61-82.

[56] E. J. Carr, I. W. Turner, P. Perré, A variable-stepsize Jacobian-free exponential integrator for simulating transport in heterogeneous porous media: Application to wood drying, J. Comput. Phys. 233 (2013) 66-82.

[57] I. Podlubny, Fractional differential equations, Academic Press, San Diego (1999).

[58] A. Bueno-Orovio, D. Kay, V. Grau, B. Rodriguez, K. Burrage, Fractional diffusion models of cardiac electrical propagation: role of structural heterogeneity in dispersion of repolarization, J. R. Soc. Interface 11 (2014) 20140352. http://dx.doi.org/10.1098/rsif.2014.0352.

[59] H. A. Van der Vorst, Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 13 (1992) 631-644.