A Numerical Algorithm Based on Quadratic Finite Element for Two-Dimensional Nonlinear Time Fractional Thermal Diffusion Model

Yanlong Zhang[1], Baoli Yin[1], Yue Cao[1], Yang Liu[1,*] and Hong Li[1]

Abstract: In this article, a high-order scheme, which is formulated by combining the quadratic finite element method in space with a second-order time discrete scheme, is developed for looking for the numerical solution of a two-dimensional nonlinear time fractional thermal diffusion model. The time Caputo fractional derivative is approximated by using the $L2\text{-}1_\sigma$ formula, the first-order derivative and nonlinear term are discretized by some second-order approximation formulas, and the quadratic finite element is used to approximate the spatial direction. The error accuracy $O(h^3 + \Delta t^2)$ is obtained, which is verified by the numerical results.

Keywords: Quadratic finite element, two-dimensional nonlinear time fractional thermal diffusion model, $L2\text{-}1_\sigma$ formula.

1 Introduction

Fractional differential equations (FDEs) are important mathematical models, which can be applied widely in sciences and engineering, such as physics, mechanics, biology, medicine, control theory signal and image processing. So, many scholars studied their analytical and numerical solutions. The advantage of FDEs compared with integer order differential equations is that they can better simulate some physical processes and dynamic system processes in nature. However most of FDEs are difficult to solve or the analytical solutions are so complex that can not be expressed by simple functions. Therefore the study of numerical methods for FDEs is meaningful in practice.

Up to now numerical methods for FDEs based on different definitions of fractional derivatives include Riemann-Liouville fractional derivative [Liu, Liu, Li et al. (2018); Li, Zhang and Zhang (2018); Wang and Du (2013); Zeng, Li, Liu et al. (2013); Abbaszadeh and Dehghan (2015); Meerschaert and Tadjeran (2004); Liu, Zhang, Li et al. (2016); Wang, Liu, Li et al. (2016); Liu, Du, Li et al. (2019)], Riesz fractional derivative [Feng, Zhuang, Liu et al. (2015, 2016); Yin, Liu, Li et al. (2019); Cheng, Duan and Li (2019)], Grümwald-

[1]School of Mathematical Sciences, Inner Mongolia University, Hohhot, 010021, China.

*Corresponding Author: Yang Liu. Email: mathliuyang@imu.edu.cn.

Received: 22 July 2019; Accepted: 20 August 2019.

Letnikov fractional derivative [Li, Huang and Lin (2011)] and Caputo fractional derivative [Liu, Du, Li et al. (2015a,b); Zhao, Zhang, Liu et al. (2016); Chen and Li (2017); Liu, Yu, Li et al. (2018); Yuste and Quintana-Murillo (2012); Liu, Zheng, Chen et al. (2018); Heydari, Hooshmandasl and Mohammadi (2014); Vong, Lyu and Wang (2016); Zhang, Sun and Wu (2011); Feng, Liu and Turner (2019); Li, Wu and Zhang (2019); Liao, Yan and Zhang (2019); Lyu and Vong (2018)], and so forth. In this article, we consider the nonlinear time fractional thermal diffusion model with the Caputo fractional derivative. Several approximation formulas have been developed for this fractional derivative. In Sun et al. [Sun and Wu (2006); Lin and Xu (2007)], the $L1$ method was proposed with the convergence order $(2-\alpha)$ $(0<\alpha<1)$. In Gao et al. [Gao, Sun and Sun (2015)], Gao et al. proposed the $L1$-2 formula to approximate the Caputo fractional derivative. According to the idea of the literature [Gao, Sun and Sun (2015)], Alikhanov [Alikhanov (2015)] constructed an $L2$-1_σ formula to approximate the Caputo fractional derivative with error accuracy $O(\tau^{3-\alpha})$.

In our study we develop the quadratic finite element method with the $L2$-1_σ formula to solve the following two-dimensional nonlinear time fractional thermal diffusion model

$$\frac{\partial u}{\partial t}-\frac{\partial^\alpha \triangle u}{\partial t^\alpha}-\triangle u=f(u)+g(\mathbf{x},t),(\mathbf{x},t)\in\Omega\times J, \tag{1}$$

with the boundary condition

$$u(\mathbf{x},t)=0,(\mathbf{x},t)\in\partial\Omega\times\bar{J}, \tag{2}$$

and the initial condition

$$u(\mathbf{x},0)=u_0(\mathbf{x}),\mathbf{x}\in\Omega, \tag{3}$$

where $\mathbf{x}=(x_1,x_2)\in\Omega\subset R^2, J=(0,T], T\in(0,+\infty)$, $\triangle u=\frac{\partial^2 u}{\partial x_1^2}+\frac{\partial^2 u}{\partial x_2^2}$, $f(u)$ is a nonlinear term, and $\frac{\partial^\alpha w(\mathbf{x},t)}{\partial t^\alpha}$ is defined by the following Caputo fractional derivative

$$\frac{\partial^\alpha w(\mathbf{x},t)}{\partial t^\alpha}=\frac{1}{\Gamma(1-\alpha)}\int_0^t\frac{\partial w(\mathbf{x},\tau)}{\partial\tau}\frac{d\tau}{(t-\tau)^\alpha},0<\alpha<1. \tag{4}$$

We note that the quadratic finite element method has a higher error accuracy $O(h^3)$ compared with the linear finite element method. And for the temporal direction, we apply the $L2$-1_σ formula to the Caputo fractional derivative which has convergence order $3-\alpha$, and use some second-order approximation formulas for the first-order derivative and nonlinear term at $t_{n-\frac{\alpha}{2}}$. Therefore, the error accuracy $O(h^3+\Delta t^2)$ of the scheme is desired.

The rest of the paper is structured as follows. In Section 2, we present the fully discrete scheme of the equation by the quadratic finite element method in spatial direction and the $L2$-1_σ as well as some second-order formulas in temporal direction. For clarity, we offer the algorithm of our method in detail and the fully discrete scheme in matrix form is formulated. In Section 3, we conduct three numerical experiments and analyse the numerical results which confirms the efficiency of our scheme. In Section 4, we summarize the applied method and results of this paper.

2 Temporal approximation formula and quadratic finite element method

2.1 Temporal approximation formula

We divide the time interval $[0,T]$ into N equal parts with mesh length $\Delta t = T/N$, $0 = t_0 < t_1 < t_2 < \cdots < t_N = T$, and $t_n = n\Delta t (n = 0, 1, 2, \cdots, N)$. Next we give the following lemmas about approximations for nonlinear term and fractional derivative, and give specific algorithm steps.

Lemma 2.1. [Gao, Sun and Sun (2015); Wang, Liu, Li et al. (2016)] Assume $z(t) \in C^3[0,T]$. The following second-order formula for approximating the time first-order derivative at time $t_{n-\frac{\alpha}{2}}$ holds,

Case 1: $n = 1$,

$$\frac{\partial z}{\partial t}(t_{1-\frac{\alpha}{2}}) = \frac{1}{\Delta t}(z^1 - z^0) + O(\Delta t), \tag{5}$$

Case 2: $n \geq 2$,

$$\frac{\partial z}{\partial t}(t_{n-\frac{\alpha}{2}}) = \frac{1}{2\Delta t}\left[(3-\alpha)z^n - (4-2\alpha)z^{n-1} + (1-\alpha)z^{n-2}\right] + O(\Delta t^2). \tag{6}$$

Lemma 2.2. [Wang, Liu, Li et al. (2016)] Assume $f(t)$ and $g(t)$ both belong to the space $C^2[0,T]$. The following second-order formulas for approximating the source term and the nonlinear term at time $t_{n-\frac{\alpha}{2}}$ holds,

$$g(t_{n-\frac{\alpha}{2}}) = (1-\frac{\alpha}{2})g^n + (\frac{\alpha}{2})g^{n-1} + O(\Delta t^2) = g^{n-\frac{\alpha}{2}} + O(\Delta t^2), \tag{7}$$

$$\begin{aligned} f(z(t_{n-\frac{\alpha}{2}})) &= (2-\frac{\alpha}{2})f(z^{n-1}) - (1-\frac{\alpha}{2})f(z^{n-2}) + O(\Delta t^2) \\ &= f(z^{n-\frac{\alpha}{2}}) + O(\Delta t^2). \end{aligned} \tag{8}$$

Lemma 2.3. [Alikhanov (2015)] Assuming $z(t) \in C^3[0,T]$, the following $L2\text{-}1_\sigma$ formula for approximating the time Caputo fractional derivative at time $t_{n-\frac{\alpha}{2}}$ holds,

$$\frac{\partial^\alpha z}{\partial t^\alpha}(t_{n-\frac{\alpha}{2}}) = \frac{\Delta t^{-\alpha}}{\Gamma(2-\alpha)}\left[c_0^{(n)}z^n - \sum_{j=1}^{n-1}(c_{n-j-1}^{(n)} - c_{n-j}^{(n)})z^j - c_{n-1}^{(n)}z^0\right] + O(\Delta t^{3-\alpha}), \tag{9}$$

where for the case $n = 1$, denote

$$c_0^{(1)} = a_0, \tag{10}$$

and for the case $n \geq 2$, denote

$$c_q^{(n)} = \begin{cases} a_0 + b_1, & q = 0, \\ a_q + b_{q-1} - b_q, & 1 \leq q \leq n-2, \\ a_q - b_q, & q = n-1. \end{cases} \tag{11}$$

where

$$a_0 = (1-\frac{\alpha}{2})^{1-\alpha}, a_p = (p+1-\frac{\alpha}{2})^{1-\alpha} - (p-\frac{\alpha}{2})^{1-\alpha}, p \geq 1, \tag{12}$$

$$b_p = \frac{1}{2-\alpha}\left[(p+1-\frac{\alpha}{2})^{2-\alpha} - (p-\frac{\alpha}{2})^{2-\alpha}\right] - \frac{1}{2}\left[(p+1-\frac{\alpha}{2})^{1-\alpha} + (p-\frac{\alpha}{2})^{1-\alpha}\right], p \geq 1. \tag{13}$$

According to lemmas 2.1-2.3, we get the following time discrete formulation of Eq. (1) at time $t_{n-\frac{\alpha}{2}}$

Case 1: $n = 1$,

$$\frac{1}{\Delta t}(u^1 - u^0) - \frac{c_0^{(1)}\Delta t^{-\alpha}}{\Gamma(2-\alpha)}(\Delta u^1 - \Delta u^0) - \Delta u^{1-\frac{\alpha}{2}} = f(u^0) + g^{1-\frac{\alpha}{2}} + R^{1-\frac{\alpha}{2}}, \tag{14}$$

Case 2: $n \geq 2$,

$$\frac{1}{2\Delta t}\big[(3-\alpha)u^n - (4-2\alpha)u^{n-1} + (1-\alpha)u^{n-2}\big] - \frac{\Delta t^{-\alpha}}{\Gamma(2-\alpha)}\Bigg[c_0^{(n)}\Delta u^n - \sum_{j=1}^{n-1}(c_{n-j-1}^{(n)} - c_{n-j}^{(n)})\Delta u^j - c_{n-1}^{(n)}\Delta u^0\Bigg] - \Delta u^{n-\frac{\alpha}{2}} = f(u^{n-\frac{\alpha}{2}}) + g^{n-\frac{\alpha}{2}} + R^{n-\frac{\alpha}{2}}. \tag{15}$$

Further, we have the weak formulation as follows:

Case 1: $n = 1$,

$$\left(\frac{u^1 - u^0}{\Delta t}, v\right) + \frac{c_0^{(1)}\Delta t^{-\alpha}}{\Gamma(2-\alpha)}(\nabla u^1 - \nabla u^0, \nabla v) + (\nabla u^{1-\frac{\alpha}{2}}, \nabla v) = (f(u^0), v) + (g^{1-\frac{\alpha}{2}}, v) + (R^{1-\frac{\alpha}{2}}, v), \forall v \in H_0^1, \tag{16}$$

Case 2: $n \geq 2$,

$$\left(\frac{1}{2\Delta t}\big[(3-\alpha)u^n - (4-2\alpha)u^{n-1} + (1-\alpha)u^{n-2}\big], v\right) + \frac{\Delta t^{-\alpha}}{\Gamma(2-\alpha)}\left(\Bigg[c_0^{(n)}\nabla u^n - \sum_{j=1}^{n-1}(c_{n-j-1}^{(n)} - c_{n-j}^{(n)})\nabla u^j - c_{n-1}^{(n)}\nabla u^0\Bigg], \nabla v\right) + (\nabla u^{n-\frac{\alpha}{2}}, \nabla v) = (f(u^{n-\frac{\alpha}{2}}), v) + (g^{n-\frac{\alpha}{2}}, v) + (R^{n-\frac{\alpha}{2}}, v), \forall v \in H_0^1. \tag{17}$$

By choosing a finite element space $V_h \subset H_0^1$, and assuming the solution satisfies $u \in C([0,T]; H^3(\Omega) \cap H_0^1(\Omega)) \cap C^3([0,T]; L^2(\Omega))$, we can get the following fully discrete system

Case 1: $n = 1$,

$$\left(\frac{1}{\Delta t}(u_h^1 - u_h^0), v_h\right) + \frac{c_0^{(1)}\Delta t^{-\alpha}}{\Gamma(2-\alpha)}(\nabla u_h^1 - \nabla u_h^0, \nabla v_h) + (1-\frac{\alpha}{2})(\nabla u_h^1, \nabla v_h) + \frac{\alpha}{2}(\nabla u_h^0, \nabla v_h) = (f(u_h^0), v_h) + (1-\frac{\alpha}{2})(g^1, v_h) + \frac{\alpha}{2}(g^0, v_h), \forall v_h \in V_h, \tag{18}$$

Case 2: $n \geq 2$,

$$\begin{aligned}
&\left(\frac{1}{2\Delta t}\left[(3-\alpha)u_h^n-(4-2\alpha)u_h^{n-1}+(1-\alpha)u_h^{n-2}\right],v_h\right)\\
&+\frac{\Delta t^{-\alpha}}{\Gamma(2-\alpha)}\left(\left[c_0^{(n)}\nabla u_h^n-\sum_{j=1}^{n-1}(c_{n-j-1}^{(n)}-c_{n-j}^{(n)})\nabla u_h^j-c_{n-1}^{(n)}\nabla u_h^0\right],\nabla v_h\right)\\
&+(1-\frac{\alpha}{2})(\nabla u_h^n,\nabla v_h)+\frac{\alpha}{2}(\nabla u_h^{n-1},\nabla v_h)=(2-\frac{\alpha}{2})(f(u_h^{n-1}),v_h)\\
&-(1-\frac{\alpha}{2})(f(u_h^{n-2}),v_h)+(1-\frac{\alpha}{2})(g^n,v_h)+\frac{\alpha}{2}(g^{n-1},v_h),\forall v_h\in V_h.
\end{aligned} \tag{19}$$

Simplify the process (18)-(19) to arrive at

Case 1: $n = 1$,

$$\begin{aligned}
&(u_h^1,v_h)+\left(\frac{c_0^{(1)}\Delta t^{1-\alpha}}{\Gamma(2-\alpha)}+\Delta t(1-\frac{\alpha}{2})\right)(\nabla u_h^1,\nabla v_h)=\Delta t(f(u_h^0),v_h)\\
&+\Delta t(1-\frac{\alpha}{2})(g^1,v_h)+\Delta t\frac{\alpha}{2}(g^0,v_h)+(u_h^0,v_h)-\Delta t\frac{\alpha}{2}(\nabla u_h^0,\nabla v_h),\forall v_h\in V_h,
\end{aligned} \tag{20}$$

Case 2: $n \geq 2$,

$$\begin{aligned}
&\big((3-\alpha)u_h^n,v_h\big)+\left(\frac{2c_0^{(n)}\Delta t^{1-\alpha}}{\Gamma(2-\alpha)}+2\Delta t(1-\frac{\alpha}{2})\right)(\nabla u_h^n,\nabla v_h)=2\Delta t(2-\frac{\alpha}{2})\\
&(f(u_h^{n-1}),v_h)-2\Delta t(1-\frac{\alpha}{2})(f(u_h^{n-2}),v_h)+2\Delta t(1-\frac{\alpha}{2})(g^n,v_h)+\Delta t\alpha(g^{n-1},v_h)\\
&+\big((4-2\alpha)u_h^{n-1}-(1-\alpha)u_h^{n-2},v_h\big)+\frac{2\Delta t^{1-\alpha}}{\Gamma(2-\alpha)}\left(\left[\sum_{j=1}^{n-1}(c_{n-j-1}^{(n)}+c_{n-j}^{(n)})\nabla u_h^j\right.\right.\\
&\left.\left.-c_{n-1}^{(n)}\nabla u_h^0\right],\nabla v_h\right)-\Delta t\alpha(\nabla u_h^{n-1},\nabla v_h),\forall v_h\in V_h.
\end{aligned} \tag{21}$$

For the finite element system (20)-(21), we can consider the stability and error estimate in the future. Here, we merely carry out the numerical calculation for the Eq. (1). To obtain a higher order convergence rate in space, we apply quadratic finite element method to the 2D nonlinear time fractional thermal diffusion model, which is seldom explored by scholars.

2.2 Quadratic finite element method

For the 2D model defined by (1), let $\Omega = (I_1, I_2)^2 \subset R^2$, $J = (0, T]$. The space domain Ω is triangulated into shape regular and qusi-uniform partitions with the number of triangles denoted as M. Since positions of these triangles are different, they can be classified by $e^{(I)}, e^{(II)}$ as Fig. 1. For any quadratic triangular element $e = \triangle P_1P_2P_3$, we denote three vertices of the triangle e as P_1, P_2, P_3, and the midpoints on the three sides as P_4, P_5, P_6, see Fig. 2.

Then the interpolation on the triangle e is performed by using quadratic polynomial and the six coefficients of the bivariate complete polynomial $L(x, y) = a_1x^2 + a_2xy + a_3y^2 +$

$a_4x + a_5y + a_6$ are determined by the quadratic triangular element. By using the shape functions $N_i(x, y)(1 \leq i \leq 6)$, the bivariate polynomial can be expressed as $L(x, y) = \sum_{i=1}^{6}(L_i N_i(x, y))$ where $L_i = L(P_i)$.

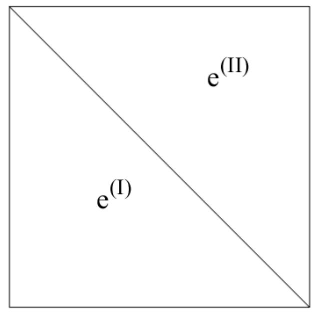

Figure 1: Triangulation

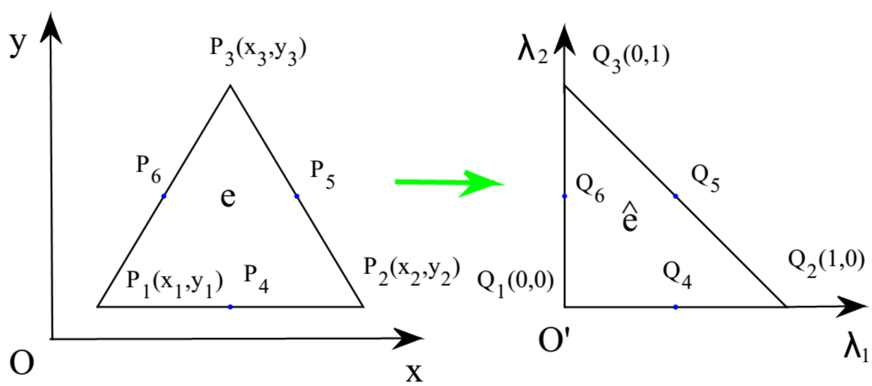

Figure 2: Affine mapping

For simplicity, we make the affine mapping to transform the element e on the (x, y) plane into the standard element $\hat{e}$ on the (λ_1, λ_2) plane, as is illustrated in Fig. 2, such that

$$\begin{cases} \lambda_1 = \frac{1}{2\Delta_e}(m_1x + m_2y + m_3), \\ \lambda_2 = \frac{1}{2\Delta_e}(n_1x + n_2y + n_3), \end{cases} \tag{22}$$

where

$$\begin{aligned} m_1 &= \begin{vmatrix} y_2 & 1 \\ y_3 & 1 \end{vmatrix}, \quad m_2 = -\begin{vmatrix} x_2 & 1 \\ x_3 & 1 \end{vmatrix}, \quad m_3 = \begin{vmatrix} x_2 & y_2 \\ x_3 & y_3 \end{vmatrix}, \\ n_1 &= \begin{vmatrix} y_3 & 1 \\ y_1 & 1 \end{vmatrix}, \quad n_2 = -\begin{vmatrix} x_3 & 1 \\ x_1 & 1 \end{vmatrix}, \quad n_3 = \begin{vmatrix} x_3 & y_3 \\ x_1 & y_1 \end{vmatrix}, \end{aligned} \tag{23}$$

and Δ_e denotes the area of triangle e defined by

$$\Delta_e = \frac{1}{2}\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}. \tag{24}$$

With the transform (22), the shape functions $N_i(x, y)(1 \leq i \leq 6)$ can be converted to the following forms, for $\lambda_1 + \lambda_2 + \lambda_3 = 1$,

$$\begin{cases} \hat{N}_1(\lambda_1, \lambda_2, \lambda_3) = \lambda_1(2\lambda_1 - 1), \hat{N}_2(\lambda_1, \lambda_2, \lambda_3) = \lambda_2(2\lambda_2 - 1), \\ \hat{N}_3(\lambda_1, \lambda_2, \lambda_3) = \lambda_3(2\lambda_3 - 1), \hat{N}_4(\lambda_1, \lambda_2, \lambda_3) = 4\lambda_2\lambda_3, \\ \hat{N}_5(\lambda_1, \lambda_2, \lambda_3) = 4\lambda_1\lambda_3, \hat{N}_6(\lambda_1, \lambda_2, \lambda_3) = 4\lambda_1\lambda_2. \end{cases} \tag{25}$$

With the affine mapping, any integral on the element e can be performed on the standard element $\hat{e}$ as follows:

$$\begin{aligned} \iint_e F(x, y)dxdy &= \iint_{\hat{e}} F(\lambda_1, \lambda_2)\Big|\frac{\partial(x, y)}{\partial(\lambda_1, \lambda_2)}\Big|d\lambda_1 d\lambda_2 \\ &= 2\Delta_e \int_0^1 d\lambda_1 \int_0^{1-\lambda_1} F(\lambda_1, \lambda_2)d\lambda_2. \end{aligned} \tag{26}$$

Similarly, we calculate the integrals of Eqs. (20) and (21) by summing integrals on each triangular element $e_l(1 \leq l \leq M)$, and then, calculate integrals on the standard element $\hat{e}$ by the following formulas

$$\iint_e u_h v_h dxdy = 2\Delta_e \sum_{i=1}^{6} v_i \sum_{j=1}^{6} u_j \iint_{\hat{e}} \hat{N}_i \hat{N}_j d\lambda_1 d\lambda_2, \tag{27}$$

and

$$\begin{aligned} &\iint_e \nabla u_h \nabla v_h dxdy = 2\Delta_e \sum_{i=1}^{6} v_i \sum_{j=1}^{6} u_j \iint_{\hat{e}} \left(\frac{\partial \hat{N}_j}{\partial x}\frac{\partial \hat{N}_i}{\partial x} + \frac{\partial \hat{N}_j}{\partial y}\frac{\partial \hat{N}_i}{\partial y}\right)d\lambda_1 d\lambda_2 \\ &= \frac{1}{2\Delta_e}\sum_{i=1}^{6} v_i \sum_{j=1}^{6} u_j \iint_{\hat{e}} \left[\left(m_1\frac{\partial \hat{N}_j}{\partial \lambda_1} + n_1\frac{\partial \hat{N}_j}{\partial \lambda_2}\right)\left(m_1\frac{\partial \hat{N}_i}{\partial \lambda_1} + n_1\frac{\partial \hat{N}_i}{\partial \lambda_2}\right)\right. \\ &\left.+ \left(m_2\frac{\partial \hat{N}_j}{\partial \lambda_1} + n_2\frac{\partial \hat{N}_j}{\partial \lambda_2}\right)\left(m_2\frac{\partial \hat{N}_i}{\partial \lambda_1} + n_2\frac{\partial \hat{N}_i}{\partial \lambda_2}\right)\right]d\lambda_1 d\lambda_2. \end{aligned} \tag{28}$$

To give the matrix form of (20) and (21), we introduce the basis functions of finite space V_h as $\mathcal{N}_1, \mathcal{N}_2, \cdots, \mathcal{N}_m$. Then the numerical solution $u_h \in V_h$ can be expressed by $u_h = (\mathcal{N}_1, \mathcal{N}_2, \cdots, \mathcal{N}_m)\mathbf{u}_h$, where the vector $\mathbf{u}_h$ is $(u_1, u_2, \cdots, u_m)^T$. Now we can rewrite Eqs. (20) and (21) as:

Case 1: $n = 1$,

$$\mathbf{M}\mathbf{u}_h^1 + \left[\frac{c_0^{(1)}\Delta t^{1-\alpha}}{\Gamma(2-\alpha)} + \Delta t(1 - \frac{\alpha}{2})\right]\mathbf{A}\mathbf{u}_h^1 = \Delta t\mathbf{M}f(\mathbf{u}_h^0) + \Delta t\mathbf{g}^1 + \mathbf{M}\mathbf{u}_h^0 - \Delta t\frac{\alpha}{2}\mathbf{A}\mathbf{u}_h^0, \tag{29}$$

Case 2: $n \geq 2$,

$$
\begin{aligned}
&(3-\alpha)\mathbf{M}\mathbf{u}_h^n + \left[\frac{2c_0^{(n)}\Delta t^{1-\alpha}}{\Gamma(2-\alpha)} + 2\Delta t(1-\frac{\alpha}{2})\right]\mathbf{A}\mathbf{u}_h^n = 2\Delta t\mathbf{M}\left[(2-\frac{\alpha}{2})f(\mathbf{u}_h^{n-1})\right.\\
&\left.-(1-\frac{\alpha}{2})f(\mathbf{u}_h^{n-2})\right] + 2\Delta t\left[(1-\frac{\alpha}{2})\mathbf{g}^n + \frac{\alpha}{2}\mathbf{g}^{n-1}\right] + (4-2\alpha)\mathbf{M}\mathbf{u}_h^{n-1} - (1-\alpha)\mathbf{M}\mathbf{u}_h^{n-2}\\
&+\frac{2\Delta t^{1-\alpha}}{\Gamma(2-\alpha)}\mathbf{A}\left[\sum_{j=1}^{n-1}(c_{n-j-1}^{(n)} + c_{n-j}^{(n)})\mathbf{u}_h^j - c_{n-1}^{(n)}\mathbf{u}_h^0\right] - \Delta t\alpha\mathbf{A}\mathbf{u}_h^{n-1},
\end{aligned}
\tag{30}
$$

where the stiffness matrix $\mathbf{A}$, mass matrix $\mathbf{M}$ and force vector $\mathbf{g}^n$ are defined by

$$\mathbf{A} = \sum_{i=1}^{m}\sum_{j=1}^{m}\iint_\Omega\left(\frac{\partial\mathcal{N}_j}{\partial x}\frac{\partial\mathcal{N}_i}{\partial x} + \frac{\partial\mathcal{N}_j}{\partial y}\frac{\partial\mathcal{N}_i}{\partial y}\right)dxdy = \sum_{l=1}^{M/2}A_l^{(I)} + \sum_{l=M/2+1}^{M}A_l^{(II)}, \tag{31}$$

$$\mathbf{M} = \sum_{i=1}^{m}\sum_{j=1}^{m}\iint_\Omega\mathcal{N}_j\mathcal{N}_i dxdy = \sum_{l=1}^{M}M_l, \tag{32}$$

$$\mathbf{g}^n = \sum_{i=1}^{m}\iint_\Omega g_i^n\mathcal{N}_i dxdy, \tag{33}$$

where $A_l^{(I)}, A_l^{(II)}, M_l$ are expanded by the element matrices $\bar{A}_l^{(I)}, \bar{A}_l^{(II)}, \bar{M}_l$ defined as follows:

$$\bar{A}_l^{(I)} = \frac{2(I_2-I_1)^2}{M}\sum_{i=1}^{6}\sum_{j=1}^{6}\iint_{e_l^{(I)}}\left(\frac{\partial N_j}{\partial x}\frac{\partial N_i}{\partial x} + \frac{\partial N_j}{\partial y}\frac{\partial N_i}{\partial y}\right)dxdy, \tag{34}$$

$$\bar{A}_l^{(II)} = \frac{2(I_2-I_1)^2}{M}\sum_{i=1}^{6}\sum_{j=1}^{6}\iint_{e_l^{(II)}}\left(\frac{\partial N_j}{\partial x}\frac{\partial N_i}{\partial x} + \frac{\partial N_j}{\partial y}\frac{\partial N_i}{\partial y}\right)dxdy, \tag{35}$$

$$\bar{M}_l = \sum_{i=1}^{6}\sum_{j=1}^{6}\iint_{e_l}N_jN_i dxdy, \tag{36}$$

where $N_i(1 \leq i \leq 6)$ are shape functions of the triangle element e. Careful calculation shows that

$$\bar{A}_l^{(I)} = (I_2-I_1)^2\begin{pmatrix}
\frac{1}{2} & \frac{1}{6} & 0 & 0 & 0 & -\frac{2}{3}\\
\frac{1}{6} & 1 & \frac{1}{6} & -\frac{2}{3} & 0 & -\frac{2}{3}\\
0 & \frac{1}{6} & \frac{1}{2} & -\frac{2}{3} & 0 & 0\\
0 & -\frac{2}{3} & -\frac{2}{3} & \frac{8}{3} & -\frac{4}{3} & 0\\
0 & 0 & 0 & -\frac{4}{3} & \frac{8}{3} & -\frac{4}{3}\\
-\frac{2}{3} & -\frac{2}{3} & 0 & 0 & -\frac{4}{3} & \frac{8}{3}
\end{pmatrix}, \tag{37}$$

$$\bar{A}_l^{(II)} = (I_2 - I_1)^2 \begin{pmatrix} \frac{1}{2} & 0 & \frac{1}{6} & 0 & -\frac{2}{3} & 0 \\ 0 & \frac{1}{2} & \frac{1}{6} & -\frac{2}{3} & 0 & 0 \\ \frac{1}{6} & \frac{1}{6} & 1 & -\frac{2}{3} & -\frac{2}{3} & 0 \\ 0 & -\frac{2}{3} & -\frac{2}{3} & \frac{8}{3} & 0 & -\frac{4}{3} \\ -\frac{2}{3} & 0 & -\frac{2}{3} & 0 & \frac{8}{3} & -\frac{4}{3} \\ 0 & 0 & 0 & -\frac{4}{3} & -\frac{4}{3} & \frac{8}{3} \end{pmatrix}, \tag{38}$$

$$\bar{M}_l = \frac{2}{M} \begin{pmatrix} \frac{1}{60} & -\frac{1}{360} & -\frac{1}{360} & -\frac{1}{90} & 0 & 0 \\ -\frac{1}{360} & \frac{1}{60} & -\frac{1}{360} & 0 & -\frac{1}{90} & 0 \\ -\frac{1}{360} & -\frac{1}{360} & \frac{1}{60} & 0 & 0 & -\frac{1}{90} \\ -\frac{1}{90} & 0 & 0 & \frac{4}{45} & \frac{2}{45} & \frac{2}{45} \\ 0 & -\frac{1}{90} & 0 & \frac{2}{45} & \frac{4}{45} & \frac{2}{45} \\ 0 & 0 & -\frac{1}{90} & \frac{2}{45} & \frac{2}{45} & \frac{4}{45} \end{pmatrix}. \tag{39}$$

Then by the algorithm (29)-(33), we can calculate the numerical solution at each time level by solving the linearized equations.

3 Numerical tests

In this section, we use the algorithm (29)-(33) to solve three numerical examples. The errors and orders of convergence are obtained by Matlab programs.

3.1 Example 1

In Eq. (1), we choose the space-time domain $\bar{\Omega} \times \bar{J} = [0,1]^2 \times [0,1]$, the nonlinear term $f(u) = u^2$ and the source term

$$\begin{aligned} g(\mathbf{x},t) =& \left[(3+\alpha)t^{2+\alpha} + 8\pi^2 t^{3+\alpha} + \frac{4\pi^2 t^3}{3}\Gamma(4+\alpha)\right]\sin(2\pi x_1)\sin(2\pi x_2) \\ &- t^{6+2\alpha}\sin^2(2\pi x_1)\sin^2(2\pi x_2), \end{aligned} \tag{40}$$

where $\mathbf{x} = (x_1, x_2)$. The exact solution to the equation is

$$u(\mathbf{x},t) = t^{3+\alpha}\sin(2\pi x_1)\sin(2\pi x_2). \tag{41}$$

We take the time step length $\Delta t = 1/N$ and the space step length $h = h_{x_1} = h_{x_2} = \sqrt{2/M}$ (M is the number of triangular elements), where h_{x_1} is the step length in x-axis and h_{x_2} is the step length in y-axis. Here the error formula $\max_{1\le n\le N} ||u_h^n - u(t_n)||$ is used to calculate errors between the numerical solution u_h and the exact solution u.
In Tab. 1, we record the errors and space convergence orders by taking $\Delta t = 1/2000$ and changed space step $h = 1/5, 1/10, 1/20$ with $\alpha = 0.1, 0.2, 0.5, 0.9$. One can see that space convergence order of the scheme is nearly 3. In Tab. 2, we fix the space step size $h = 1/40$ and set $\Delta t = 1/10, 1/20, 1/40$ with $\alpha = 0.1, 0.2, 0.5, 0.9$. One can also see that the time error accuracy $O(\Delta t^2)$ is optimal.
In addition, we compare the numerical solution and the exact solution in Figs. 3-6 at $t = 0.5$ with $\Delta t = 1/200, h = 1/20, \alpha = 0.1, 0.2, 0.5, 0.9$, respectively, where the surface represents the exact solution, and the symbol $*$ is the numerical solution at every node. Further, we depict the errors $u_h - u$ with the same setting in Figs. 7-10 which confirm the

Table 1: Errors and space convergence orders for u_h with $\Delta t = 1/2000$

α	$h_1 = 1/5$	$h_2 = 1/10$	$h_3 = 1/20$	Order$\left(\frac{h_1}{h_2}\right)$	Order$\left(\frac{h_2}{h_3}\right)$
0.1	1.5371E-02	1.9387E-03	2.4353E-04	2.98711	2.99289
0.2	1.5375E-02	1.9388E-03	2.4353E-04	2.98731	2.99299
0.5	1.5391E-02	1.9396E-03	2.4355E-04	2.98826	2.99343
0.9	1.5423E-02	1.9411E-03	2.4360E-04	2.99013	2.99427

Table 2: Errors and time convergence orders for u_h with $h = 1/40$

α	$\Delta t_1 = 1/10$	$\Delta t_2 = 1/20$	$\Delta t_3 = 1/40$	Order$\left(\frac{\Delta t_1}{\Delta t_2}\right)$	Order$\left(\frac{\Delta t_2}{\Delta t_3}\right)$
0.1	1.0882E-03	3.2870E-04	9.4333E-05	1.72707	1.80092
0.2	1.2327E-03	3.5525E-04	9.9104E-05	1.79494	1.84181
0.5	2.2145E-03	5.8736E-04	1.5380E-04	1.91468	1.93319
0.9	3.5758E-03	9.1408E-04	2.3332E-04	1.96788	1.97003

efficiency of our scheme. To further explore the distribution of errors, we draw the contour plots of $u_h - u$ in Figs. 11-12, with $\Delta t = 1/200, h = 1/40, \alpha = 0.2$ at fixed $x_1 = 0.25$ and 0.75, respectively. Similarly, we illustrate the distribution of $u_h - u$ using the same parameters with the exception that $\alpha = 0.9$ in Figs. 13-14. A direct conclusion is that the maximal error $\max_{1\le n\le N} |u_h^n - u^n|$ is yielded near $t = 1$.

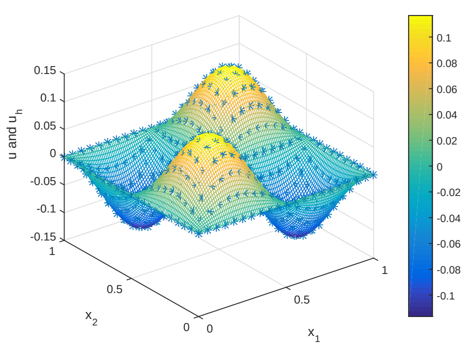

Figure 3: u and u_h at $t = 0.5$ with $\alpha = 0.1$

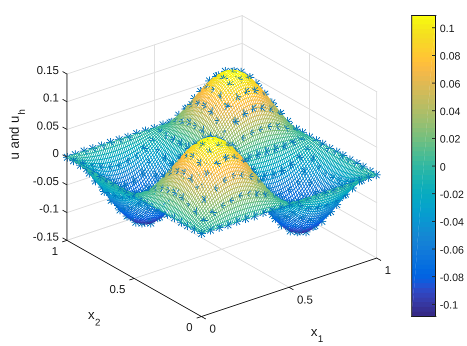

Figure 4: u and u_h at $t = 0.5$ with $\alpha = 0.2$

3.2 Example 2

Now we consider another numerical example. We take the space-time domain $\bar{\Omega} \times \bar{J} = [0, 2]^2 \times [0, 2]$, the nonlinear term $f(u) = \sin(u)$ and the source term

$$g(\mathbf{x}, t) = 3t^2 \widetilde{g}(\mathbf{x}) - 12\left[t^3 + \frac{6t^{3-\alpha}}{\Gamma(4-\alpha)}\right]\bar{g}(\mathbf{x}) - \sin(t^3 \widetilde{g}(\mathbf{x})), \tag{42}$$

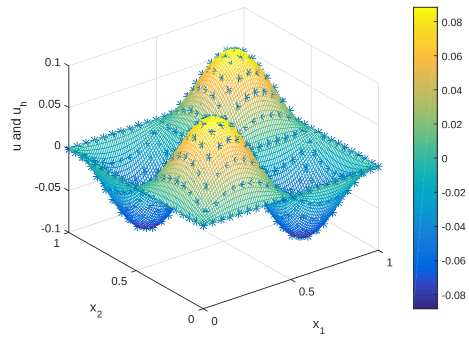

Figure 5: u and u_h at $t = 0.5$ with $\alpha = 0.5$

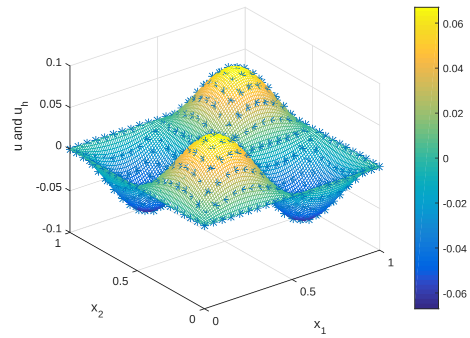

Figure 6: u and u_h at $t = 0.5$ with $\alpha = 0.9$

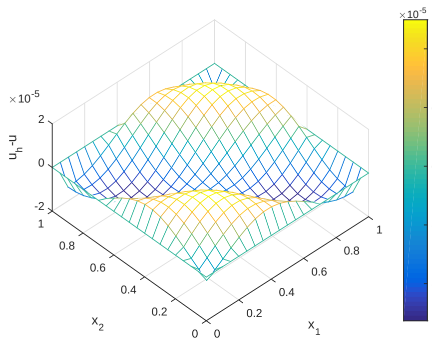

Figure 7: Error $u_h - u$ at $t = 0.5$ with $\alpha = 0.1$

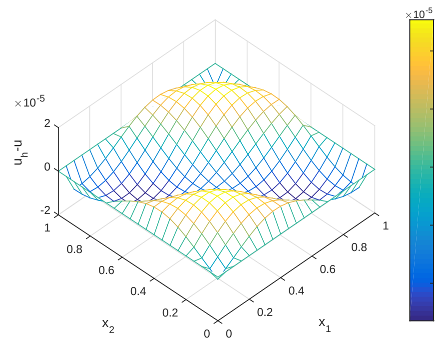

Figure 8: Error $u_h - u$ at $t = 0.5$ with $\alpha = 0.2$

$$\widetilde{g}(\mathbf{x}) = x_1(x_1 - 0.5)(x_1 - 1.5)(x_1 - 2)x_2(x_2 - 0.5)(x_2 - 1.5)(x_2 - 2), \tag{43}$$

$$\begin{aligned} \bar{g}(\mathbf{x}) =& x_1(x_1 - 0.5)(x_1 - 1.5)(x_1 - 2)\big(x_2^2 - 2x_2 + \frac{19}{24}\big) \\ &+ x_2(x_2 - 0.5)(x_2 - 1.5)(x_2 - 2)\big(x_1^2 - 2x_1 + \frac{19}{24}\big), \end{aligned} \tag{44}$$

The exact solution to the model is

$$u(\mathbf{x}, t) = t^3 x_1(x_1-0.5)(x_1-1.5)(x_1-2)x_2(x_2-0.5)(x_2-1.5)(x_2-2), \mathbf{x} = (x_1, x_2). \tag{45}$$

For checking the space error accuracy, we consider the fixed time step size $\Delta t = 1/2000$ and take the space step length $h = 1/5, 1/10, 1/20$ with $\alpha = 0.1, 0.2, 0.5, 0.9$ to calculate the errors and space convergence orders between the numerical solution u_h and the exact solution u, see Tab. 3. We find that the space error accuracy of the scheme is $O(h^3)$. Furthermore, by taking $h = 1/60$ and changing time step $\Delta t = 1/5, 1/10, 1/20$ in Tab. 4 with $\alpha = 0.1, 0.2, 0.5, 0.9$, one can see the time error accuracy of the scheme $O(\Delta t^2)$ is obtained.

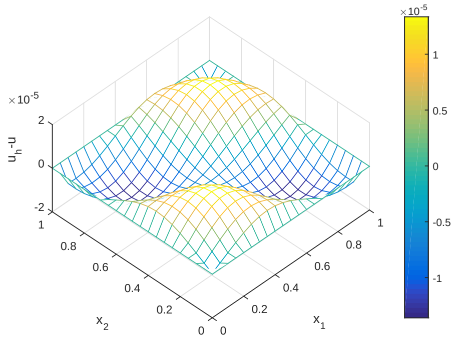

Figure 9: Error $u_h - u$ at $t = 0.5$ with $\alpha = 0.5$

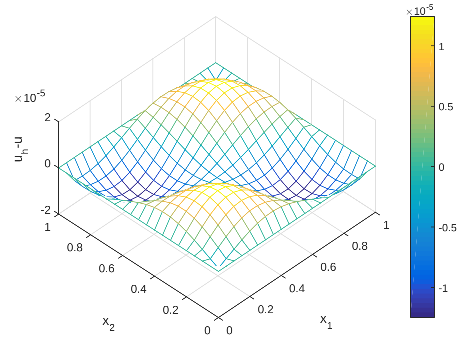

Figure 10: Error $u_h - u$ at $t = 0.5$ with $\alpha = 0.9$

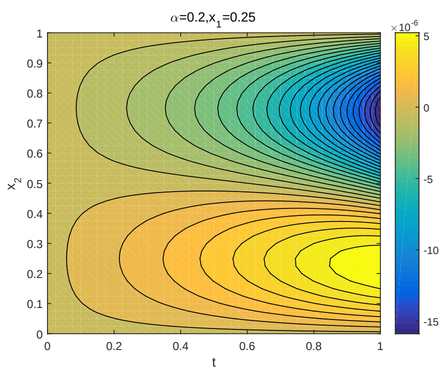

Figure 11: Error $u_h - u$ at $x_1 = 0.25$ with $\alpha = 0.2$

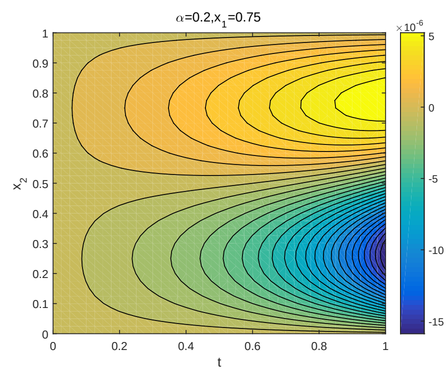

Figure 12: Error $u_h - u$ at $x_1 = 0.75$ with $\alpha = 0.2$

Table 3: Errors and space convergence orders for u_h with $\Delta t = 1/2000$

α	$h_1 = 1/5$	$h_2 = 1/10$	$h_3 = 1/20$	Order$\left(\frac{h_1}{h_2}\right)$	Order$\left(\frac{h_2}{h_3}\right)$
0.1	3.4787E-02	4.4416E-03	5.3826E-04	2.96942	3.04469
0.2	3.4786E-02	4.4415E-03	5.3826E-04	2.96938	3.04467
0.5	3.4781E-02	4.4412E-03	5.3825E-04	2.96927	3.04462
0.9	3.4776E-02	4.4410E-03	5.3823E-04	2.96915	3.04456

In Figs. 15-16, we draw the contour plots of the numerical solution u_h and the exact solution u, respectively, at $t = 1$ with $\Delta t = 1/400, h = 1/40, \alpha = 0.5$. The numerical solution and the exact solution match exactly. Further, with $\Delta t = 1/400, h = 1/40, \alpha = 0.5$, the contour plots of the errors $u_h - u$ are given in Figs. 17-20 at $t = 0.4, 0.8, 1.5, 2$. By comparing above contour plots, we find that the maximal error $|u_h - u|$ is roughly yielded near $t = 2$.

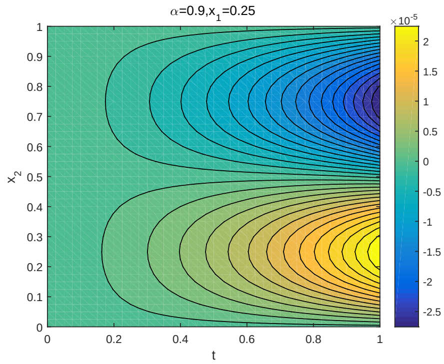

Figure 13: Error $u_h - u$ at $x_1 = 0.25$ with $\alpha = 0.9$

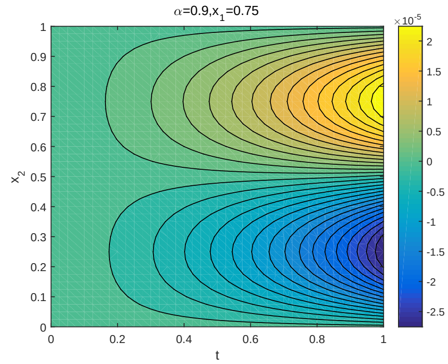

Figure 14: Error $u_h - u$ at $x_1 = 0.75$ with $\alpha = 0.9$

Table 4: Errors and time convergence orders for u_h with $h = 1/60$

α	$\Delta t_1 = 1/5$	$\Delta t_2 = 1/10$	$\Delta t_3 = 1/20$	Order$\left(\frac{\Delta t_1}{\Delta t_2}\right)$	Order$\left(\frac{\Delta t_2}{\Delta t_3}\right)$
0.1	1.2029E-03	3.3117E-04	8.5092E-05	1.86091	1.96046
0.2	1.2021E-03	3.2502E-04	8.4498E-05	1.88695	1.94356
0.5	2.2143E-03	5.7576E-04	1.4973E-04	1.94333	1.94310
0.9	2.5131E-03	6.2971E-04	1.6035E-04	1.99669	1.97350

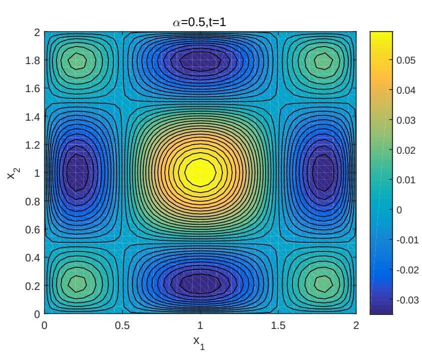

Figure 15: Numerical solution u_h at $t = 1$

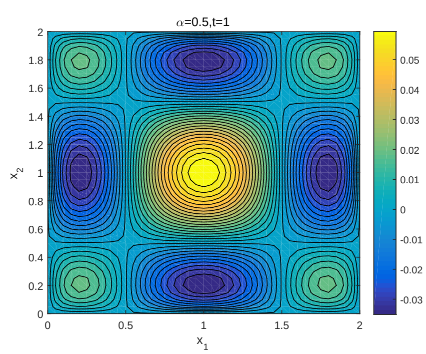

Figure 16: Exact solution u at $t = 1$

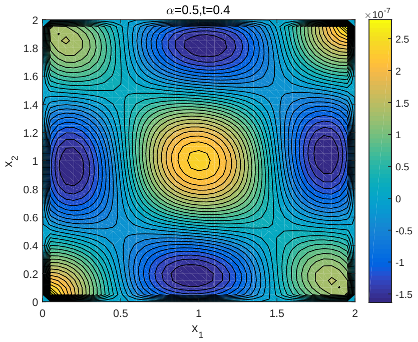

Figure 17: Error $u_h - u$ at $t = 0.4$

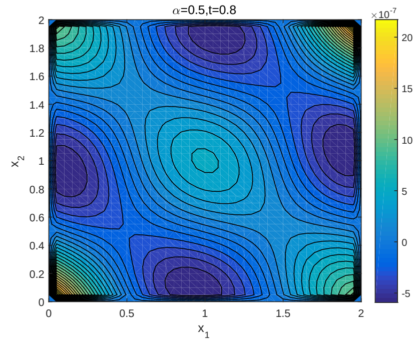

Figure 18: Error $u_h - u$ at $t = 0.8$

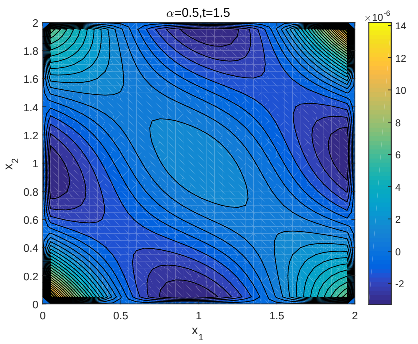

Figure 19: Error $u_h - u$ at $t = 1.5$

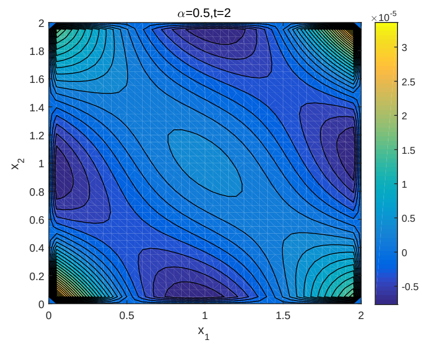

Figure 20: Error $u_h - u$ at $t = 2$

Table 5: Errors and space convergence orders for u_h with $\Delta t = 1/200$

α	$h_1 = 1/5$	$h_2 = 1/10$	$h_3 = 1/20$	Order$\left(\frac{h_1}{h_2}\right)$	Order$\left(\frac{h_2}{h_3}\right)$
0.1	1.4997E-05	1.8988E-06	2.3912E-07	2.98154	2.98933
0.2	1.6346E-05	2.0921E-06	2.6354E-07	2.96589	2.98884
0.5	1.9166E-05	2.4863E-06	3.1378E-07	2.94648	2.98620
0.9	2.0215E-05	2.6262E-06	3.3153E-07	2.94436	2.98577

3.3 Example 3

We take the nonlinear term $f(u) = u - u^3$ and the source term $g(\mathbf{x}, t) = 0$ with initial condition

$$u_0(\mathbf{x}) = x_1^2(x_1 - 1)^2 x_2^2 (x_2 - 1)^2, \mathbf{x} \in \bar{\Omega} \quad (46)$$

where $\bar{\Omega} \times \bar{J} = [0, 1]^2 \times [0, 1]$. Since the exact solution of the example is unknown, the calculated numerical solution based on the fixed time step size $\Delta t = 1/200$ and space step length $h = 1/40$ is considered as the approximate exact solution. We calculate the errors and space convergence orders between the numerical solution u_h and the exact solution u by taking $\Delta t = 1/200$ and changed space step $h = 1/5, 1/10, 1/20$, see Tab. 5. From the computing results with different $\alpha = 0.1, 0.2, 0.5, 0.9$, it is noticeable that our scheme can also effectively solve the two-dimensional nonlinear time fractional thermal diffusion model in Example 3.

4 Conclusion

In this paper, we derive the fully discrete scheme for the two-dimensional nonlinear time fractional thermal diffusion model. In temporal direction, we use the $L2\text{-}1_\sigma$ formula to approximate the time Caputo fractional derivative, and take the second-order formulas to approximate the time first-order derivative as well as the nonlinear term, at time $t_{n-\frac{\alpha}{2}}$. In spatial direction, we employ the quadratic finite element method to obtain a higher-order convergence rate. The algorithm process is given in detail and to confirm the efficiency of our scheme we conduct three numerical experiments with the help of Matlab programs. The numerical results show that the space convergence order is 3 and the time convergence order is 2, which are optimal in both directions for our scheme.

Acknowledgement: Authors thank the editor and three reviewers for their valuable comments for greatly improving this paper. This work is supported by the National Natural Science Fund (11661058, 11761053), Natural Science Fund of Inner Mongolia Autonomous Region (2016MS0102, 2017MS0107), Program for Young Talents of Science and Technology in Universities of Inner Mongolia Autonomous Region (NJYT-17-A07), National Undergraduate Innovative Training Project of Inner Mongolia University (201710126026).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study

References

Abbaszadeh, M.; Dehghan, M. (2015): A meshless numerical procedure for solving fractional reaction subdiffusion model via a new combination of alternating direction implicit (ADI) approach and interpolating element free galerkin (EFG) method. *Computers & Mathematics with Applications*, vol. 70, pp. 2493-2512.

Alikhanov, A. A. (2015): A new difference scheme for the time fractional diffusion equation. *Journal of Computational Physics*, vol. 280, pp. 424-438.

Chen, A.; Li, C. P. (2017): An alternating direction galerkin method for a time-fractional partial differential equations with damping in two space dimenstions. *Advances in Difference Equations*, vol. 2017, pp. 356.

Cheng, X. J.; Duan, J. Q.; Li, D. F. (2019): A novel compact ADI scheme for two-dimensional riesz space fractional nonlinear reaction-diffusion equations. *Applied Mathematics and Computation*, vol. 346, pp. 452-464.

Feng, L. B.; Liu, F.; Turner, I. (2019): Finite difference/finite element method for a novel 2D multi-term time-fractional mixed sub-diffusion and diffusion-wave equation on convex domains. *Communications in Nonlinear Science and Numerical Simulation*, vol. 70, pp. 354-371.

Feng, L. B.; Zhuang, P.; Liu, F.; Turner, I. (2015): Stability and convergence of a new finite volume method for a two-sided space-fractional diffusion equation. *Applied Mathematics & Computation*, vol. 257, pp. 52-65.

Feng, L. B.; Zhuang, P.; Liu, F.; Turner, I.; Gu, Y. (2016): Finite element method for space-time fractional diffusion equation. *Numerical Algorithms*, vol. 72, pp. 749-767.

Gao, G. H.; Sun, H. W.; Sun, Z. Z. (2015): Stability and convergence of finite difference schemes for a class of time-fractional sub-diffusion equations based on certain superconvergence. *Journal of Computational Physics*, vol. 280, pp. 510-528.

Heydari, M. H.; Hooshmandasl, M. R.; Mohammadi, F. (2014): Two-dimensional legendre wavelets for solving time fractional telegraph equation. *Advances in Applied Mathematics and Mechanics*, vol. 6, pp. 247-260.

Li, D. F.; Wu, C. D.; Zhang, Z. M. (2019): Linearized galerkin FEMs for nonlinear time fractional parabolic problems with non-smooth solutions in time direction. *Journal of Scientific Computing*, vol. 80, pp. 403-419.

Li, D. F.; Zhang, J. W.; Zhang, Z. M. (2018): Unconditionally optimal error estimates of a linearized galerkin method for nonlinear time fractional reaction-subdiffusion equations. *Journal of Scientific Computing*, vol. 76, no. 2, pp. 848-866.

Li, J. C.; Huang, Y. Q.; Lin, Y. P. (2011): Developing finite element methods for Maxwell's equations in a cole-cole dispersive medium. *SIAM Journal on Scientific Computing*, vol. 33, no. 6, pp. 3153-3174.

Liao, H. L.; Yan, Y. G.; Zhang, J. W. (2019): Unconditional convergence of a fast two-level linearized algorithm for semilinear subdiffusion equations. *Journal of Scientific Computing*, vol. 80, no. 1, pp. 1-25.

Lin, Y. M.; Xu, C. J. (2007): Finite difference/spectral approximations for the time-fractional diffusion equation. *Journal of Computational Physics*, vol. 225, no. 2, pp. 1533-1552.

Liu, L.; Zheng, L. C.; Chen, Y. P.; Liu, F. W. (2018): Fractional boundary layer flow and heat transfer over a stretching sheet with variable thickness. *Journal of Heat Transfer*, vol. 140, pp. 09170119.

Liu, N.; Liu, Y.; Li, H.; Wang, J. F. (2018): Time second-order finite difference/finite element algorithm for nonlinear time-fractional diffusion problem with fourth-order derivative term. *Computers & Mathematics with Applications*, vol. 75, no. 10, pp. 3521-3536.

Liu, Y.; Du, Y. W.; Li, H.; He, S.; Gao, W. (2015): Finite difference/finite element method for a nonlinear time-fractional fourth-order reaction-diffusion problem. *Computers & Mathematics with Applications*, vol. 70, no. 4, pp. 573-591.

Liu, Y.; Du, Y. W.; Li, H.; Li, J. C.; He, S. (2015): A two-grid mixed finite element method for a nonlinear fourth-order reaction-diffusion problem with time-fractional derivative. *Computers & Mathematics with Applications*, vol. 70, no. 10, pp. 2474-2492.

Liu, Y.; Du, Y. W.; Li, H.; Liu, F. W.; Wang, Y. J. (2019): Some second-order θ schemes combined with finite element method for nonlinear fractional cable equation. *Numerical Algorithms*, vol. 80, pp. 533-555.

Liu, Y.; Yu, Z. D.; Li, H.; Liu, F. W.; Wang, J. F. (2018): Time two-mesh algorithm combined with finite element method for time fractional water wave model. *International Journal of Heat and Mass Transfer*, vol. 120, pp. 1132-1145.

Liu, Y.; Zhang, M.; Li, H.; Li, J. C. (2016): High-order local discontinuous galerkin method combined with WSGD-approximation for a fractional subdiffusion equation. *Computers & Mathematics with Applications*, vol. 73, no. 6, pp. 1298-1314.

Lyu, P.; Vong, S. (2018): A linearized second-order finite difference scheme for time fractional generalized BBM equation. *Applied Mathematics Letters*, vol. 78, pp. 16-23.

Meerschaert, M. M.; Tadjeran, C. (2004): Finite difference approximations for fractional advection-dispersion flow equations. *Journal of Computational and Applied Mathematics*, vol. 172, no. 1, pp. 65-77.

Sun, Z. Z.; Wu, X. N. (2006): A fully discrete difference scheme for a diffusion-wave system. *Applied Numerical Mathematics*, vol. 56, no. 2, pp. 193-209.

Vong, S.; Lyu, P.; Wang, Z. B. (2016): A compact difference scheme for fractional sub-diffusion equations with the spatially variable coefficient under neumann boundary conditions. *Journal of Scientific Computing*, vol. 66, no. 2, pp. 725-739.

Wang, H.; Du, N. (2013): A superfast-preconditioned iterative method for steady-state space-fractional diffusion equations. *Journal of Computational Physics*, vol. 240, pp. 49-57.

Wang, Y. J.; Liu, Y.; Li, H.; Wang, J. F. (2016): Finite element method combined with second-order time discrete scheme for nonlinear fractional cable equation. *European Physical Journal Plus*, vol. 131, no. 3, pp. 1-16.

Yin, B. L.; Liu, Y.; Li, H.; He, S. (2019): Fast algorithm based on TT-M FE system for space fractional Allen-Cahn equations with smooth and non-smooth solutions. *Journal of Computational Physics*, vol. 379, pp. 351-372.

Yuste, S. B.; Quintana-Murillo, J. (2012): A finite difference method with non-uniform timesteps for fractional diffusion equations. *European Physical Journal Special Topics*, vol. 183, no. 12, pp. 2594-2600.

Zeng, F. H.; Li, C. P.; Liu, F. W.; Turner, I. (2013): The use of finite difference/element approaches for solving the time-fractional subdiffusion equation. *SIAM Journal on Scientific Computing*, vol. 35, no. 6, pp. A2976-A3000.

Zhang, Y. N.; Sun, Z. Z.; Wu, H. W. (2011): Error estimates of Crank-Nicolson-type difference schemes for the subdiffusion equation. *SIAM Journal on Numerical Analysis*, vol. 49, no. 6, pp. 2302-2322.

Zhao, Y. M.; Zhang, Y. D.; Liu, F.; Turner, I.; Tang, Y. F. et al. (2016): Convergence and superconvergence of a fully-discrete scheme for multi-term time fractional diffusion equations. *Computers & Mathematics with Applications*, vol. 73, no. 6, pp. 1087-1099.