Communications on Applied Mathematics and Computation (2020) 2:689–709
https://doi.org/10.1007/s42967-020-00065-7

ORIGINAL PAPER

A Local Discontinuous Galerkin Method for Two-Dimensional Time Fractional Diffusion Equations

Somayeh Yeganeh[1] · Reza Mokhtari[1] · Jan S. Hesthaven[2]

Received: 22 July 2019 / Revised: 10 March 2020 / Accepted: 14 March 2020 / Published online: 19 May 2020

Abstract
For two-dimensional (2D) time fractional diffusion equations, we construct a numerical method based on a local discontinuous Galerkin (LDG) method in space and a finite difference scheme in time. We investigate the numerical stability and convergence of the method for both rectangular and triangular meshes and show that the method is unconditionally stable. Numerical results indicate the effectiveness and accuracy of the method and confirm the analysis.

Keywords Two-dimensional (2D) time fractional diffusion equation · Local discontinuous Galerkin method (LDG) · Numerical stability · Convergence analysis

Mathematics Subject Classification 65M60 · 65M12

1 Introduction

Time fractional diffusion equations are obtained from the standard diffusion equations by replacing the first-order time derivative with a fractional derivative of order $\alpha \in (0, 2)$. In this paper, we consider the following time-fractional diffusion equation:

$$\begin{cases} \mathrm{D}_t^{\alpha} u(\mathrm{x}, t) - \Delta u(\mathrm{x}, t) = f(\mathrm{x}, t), & \mathrm{x} = (x, y) \in \Omega, \ t \in (0, T], \\ u|_{t=0} = \Phi(x, y), & (x, y) \in \Omega, \\ u(x, y, t) = \Psi(x, y, t), & (x, y) \in \partial\Omega, \quad t \in (0, T], \end{cases} \tag{1}$$

where Φ, Ψ, and f are given functions, $\Omega \subset \mathbb{R}^2$ is a bounded rectangular domain with boundary $\partial\Omega$, and D_t^{α} is the Caputo fractional derivative defined as

✉ Reza Mokhtari
mokhtari@iut.ac.ir

[1] Department of Mathematical Sciences, Isfahan University of Technology, Isfahan 84156-83111, Iran

[2] EPFL-SB-MATHICES-MCSS, École Polytechnique Fédéral de Lausanne, 1015 Lausanne, Switzerland

$$\mathrm{D}_t^{\alpha} u(\mathrm{x}, t) = \frac{1}{\mathbf{\Gamma}(1-\alpha)} \int_0^t \frac{\partial u(\mathrm{x}, s)}{\partial s} \frac{\mathrm{d}s}{(t-s)^{\alpha}}, \quad 0 < \alpha < 1, \tag{2}$$

in which $\mathbf{\Gamma}$ is the Gamma function. For $\alpha = 1$, we have $\mathrm{D}_t^{\alpha} u = u_t$.

Diffusion equations of fractional order are used to describe anomalous diffusion in transport processes, see, e.g., [30], and monographs [31, 33]. Recently, some handbooks related to fractional calculus and its applications in physics, engineering, and life science as well as social science have been published. Interested readers find some basic theories in volume 1 [22] of the handbooks and some other materials can be found in other volumes. Analytic solutions for most fractional partial differential equations (FPDEs) are complicated to obtain or cannot be expressed explicitly. Hence, methods for finding numerical solutions for such problems are required. Numerical methods proposed for solving the FPDEs include finite difference methods [1, 6, 9, 11, 15, 25, 28, 29, 45], finite element methods [12, 14, 19, 20, 36, 44], the finite volume method [21], spectral methods [2, 24, 26], discontinuous Galerkin methods [7, 8, 13, 39, 41, 42], and homotopy and variational methods [17, 34, 40, 43].

The first local discontinuous Galerkin (LDG) method was introduced by Cockburn and Shu in [5] for time-dependent convection–diffusion systems. The LDG method is applied increasingly from the end of the twentieth century, see, e.g., [3, 4, 7, 8, 10, 13, 18, 27, 35, 37–39, 41, 42] and their bibliographies. In [10], an LDG method with both rectangular and triangular meshes for linear time-dependent fourth-order problems has been applied successfully. Recently, Huang et al. [18] applied an LDG method to solve a one-dimensional (1D) time-fractional diffusion–wave equation or super-diffusion equation, i.e., a fractional diffusion equation with $1 < \alpha < 2$. More recently, Eshaghi et al. [13] exploited piecewise tensor product polynomials of degree k on Cartesian meshes and constructed an LDG method for two-dimensional (2D) semi-linear time-fractional diffusion equations. Their method is not applicable to non-rectangular domains. While it is unconditionally stable and has order of convergence $O((\Delta t)^2 + h^{k+1/2})$, it is not optimal.

In this paper, we consider 2D time-fractional sub-diffusion equations, i.e., fractional diffusion equations with $0 < \alpha < 1$. Using the well-known L1 formula, we construct an unconditionally stable LDG method. Our method can be applied to non-rectangular domains as well. This paper is organized as follows. We provide some preliminaries in the next section. In Sect. 3, we construct an LDG method for the time fractional diffusion problem (1). The stability and convergence of the method are analyzed in Sect. 4. In Sect. 5, numerical experiments are carried out to confirm the theoretical results.

2 Notations and Some Preliminaries

We consider problems posed on the physical domain Ω with boundary $\partial\Omega$, a union of K non-overlapping local elements D^{κ}, $\kappa = 1, \cdots, K$ such that

$$\Omega \cong \Omega_h = \bigcup_{\kappa=1}^{K} D^{\kappa}.$$

Here, D^{κ} is a shape-regular triangle for triangular meshes, or a shape-regular rectangle for Cartesian meshes. We set $h_{\kappa} := \mathrm{diam}(D^{\kappa})$, $h := \max_{1 \le \kappa \le K} h_{\kappa}$, and ξ_h the set of all faces. Let Γ denote the union of the boundary faces of elements $D^{\kappa} \in \Omega_h$, in other words $\Gamma = \bigcup_{D^{\kappa} \in \Omega_h} \partial D^{\kappa}$.

Γ is composed of two parts: the set of unique internal edges Γ_i and the set of external edges $\Gamma_b = \partial\Omega$ at the domain boundaries, and $\Gamma = \Gamma_i \cup \Gamma_b$. We define the following function spaces:

$$V_h^k := \left\{ v \in L^2(\Omega_h) \middle|\ v|_{D^\kappa} \in \mathcal{R}^k(D^\kappa), \forall D^\kappa \in \Omega_h \right\},$$
$$\boldsymbol{V}_h^k := \left\{ \boldsymbol{v} \in \boldsymbol{L}^2(\Omega_h) \middle|\ \boldsymbol{v}|_{D^\kappa} \in \boldsymbol{\mathcal{R}}^k(D^\kappa), \forall D^\kappa \in \Omega_h \right\},$$

where corresponding to the triangular meshes, $\mathcal{R}^k(D^\kappa)$ is $\mathcal{P}^k(D^\kappa)$ the space of polynomials of degree at most $k \geq 0$ defined on D^κ and corresponding to the Cartesian meshes, $\mathcal{R}^k(D^\kappa)$ is $\mathcal{Q}^k(D^\kappa)$ the space of tensor product of polynomials of degrees at most k in each variable. Here, $\boldsymbol{\mathcal{R}}^k(D^\kappa) = \left(\mathcal{R}^k(D^\kappa)\right)^2$ and $\boldsymbol{L}^2(\Omega_h) = \left(L^2(\Omega_h)\right)^2$. For $v, w \in V_h^k$ and $\boldsymbol{v}, \boldsymbol{w} \in \boldsymbol{V}_h^k$, we define the following notations:

$$(v, w) := (v, w)_{\Omega_h} = \sum_{D^\kappa \in \Omega_h} (v, w)_{D^\kappa}, \quad (v, w)_{D^\kappa} := \int_{D^\kappa} v(\mathrm{x})\, w(\mathrm{x})\mathrm{dx},$$
$$(\boldsymbol{v}, \boldsymbol{w}) := (\boldsymbol{v}, \boldsymbol{w})_{\Omega_h} = \sum_{D^\kappa \in \Omega_h} (\boldsymbol{v}, \boldsymbol{w})_{D^\kappa}, \quad (\boldsymbol{v}, \boldsymbol{w})_{D^\kappa} := \int_{D^\kappa} \boldsymbol{v}(\mathrm{x}) \cdot \boldsymbol{w}(\mathrm{x})\mathrm{dx},$$
$$\langle v, \boldsymbol{v}\cdot\boldsymbol{n}\rangle := \langle v, \boldsymbol{v}\cdot\boldsymbol{n}\rangle_{\partial\Omega_h} = \sum_{D^\kappa \in \Omega_h} \langle v, \boldsymbol{v}\cdot\boldsymbol{n}\rangle_{\partial D^\kappa}, \quad \langle v, \boldsymbol{v}\cdot\boldsymbol{n}\rangle_{\partial D^\kappa} := \int_{\partial D^\kappa} v(s)\boldsymbol{v}(s)\cdot\boldsymbol{n}\mathrm{d}s,$$

where $\boldsymbol{n}$ is the outward normal unit vector to ∂D^κ. Let $H^l(\Omega_h)$ be the space of functions on Ω_h whose restriction to each element D^κ belongs to the Sobolev space $H^l(D^\kappa)$ and set $\boldsymbol{H}^l(\Omega_h) = \left(H^l(\Omega_h)\right)^2$. For any real-valued function $v \in H^l(\Omega_h)$ and any vector-valued function $\boldsymbol{v} = (v_1, v_2) \in \boldsymbol{H}^l(\Omega_h)$, we set

$$\|v\|_{H^l(\Omega_h)} := \left(\sum_{D^\kappa \in \Omega_h} \|v\|^2_{H^l(D^\kappa)} \right)^{\frac{1}{2}}, \qquad \|\boldsymbol{v}\|_{\boldsymbol{H}^l(\Omega_h)} := \left(\sum_{i=1}^{2} \|v_i\|^2_{H^l(\Omega_h)} \right)^{\frac{1}{2}}.$$

To demonstrate the flux functions, we select a fixed vector r which is not parallel to any normal at the element boundaries. This is possible because there is only a finite number of element boundary normals for any given mesh. For each face e, we apply the vector r to uniquely determine the "left" and "right" elements E_{L} and E_{R} which share the same face e. To fix the notation, for $e \in \Gamma$, we refer to the interior information of the element by a superscript "$-$" and to the exterior information by a superscript "$+$". Using this notation, we define the average

$$\{u\} = \frac{u^+ + u^-}{2} \quad \text{on} \quad e \in \Gamma_i,$$
$$\{u\} = u \quad \text{on} \quad e \in \Gamma_b,$$

where u can be a scalar or a vector. We also define the jumps along a unit normal, $\boldsymbol{n}$, as

$$[u] = \boldsymbol{n}^- u^- + \boldsymbol{n}^+ u^+, \quad [\boldsymbol{u}] = \boldsymbol{n}^- \cdot \boldsymbol{u}^- + \boldsymbol{n}^+ \cdot \boldsymbol{u}^+, \qquad \forall e \in \Gamma_i,$$
$$[u] = \boldsymbol{n} u, \quad [\boldsymbol{u}] = \boldsymbol{n} \cdot \boldsymbol{u}, \qquad \forall e \in \Gamma_b.$$

For Cartesian meshes in a multidimensional space, $\boldsymbol{P}^-$ is the tensor product of the well-known 1D projections, see, e.g., [4, 10]. In this case, the projection $\boldsymbol{P}^-$ has the following superconvergence property, see Lemma 3.7 in [10].

Lemma 1 *If* $(\eta, \boldsymbol{\rho}) \in H^{k+2}(\Omega_h) \times \boldsymbol{V}_h^k$, *we have*

$$| (\eta - \boldsymbol{P}^- \eta, \nabla \cdot \boldsymbol{\rho}) - \langle \eta - \widehat{P^- \eta}, \boldsymbol{\rho} \cdot \boldsymbol{n} \rangle_{\partial \Omega_h} | \leq C h^{k+1} \|\eta\|_{H^{k+2}(\Omega_h)} \|\boldsymbol{\rho}\|_{L^2(\Omega_h)},$$

where C depends only on k and the shape regular constant.

For triangular meshes, following [3, 10], we define the L^2-projection $\boldsymbol{P}$ for scalar-valued functions and the projection $\boldsymbol{P}^-$ is defined for vector-valued functions. More precisely, for a given function $\eta \in L^2(\Omega_h)$ and an arbitrary element $D^\kappa \in \Omega_h$, the restriction of $\boldsymbol{P}\eta$ to D^κ is defined as the element of $\mathcal{P}^k(D^\kappa)$ that satisfies

$$(\boldsymbol{P}\eta - \eta, w)_{D^\kappa} = 0, \qquad \forall w \in \mathcal{P}^k(D^\kappa). \tag{3}$$

For a given function $\boldsymbol{\rho} \in \boldsymbol{H}^1(\Omega_h)$, an arbitrary simplex $D^\kappa \in \Omega_h$, and an arbitrary edge $\tilde{e} \in \partial D^\kappa$ that satisfies $[1\ 1] \cdot \boldsymbol{n}|_{\tilde{e}} > 0$, the restriction of $\boldsymbol{P}^- \boldsymbol{\rho}$ to D^κ is defined as the element of $\boldsymbol{\mathcal{P}}^k(D^\kappa)$ that satisfies

$$\begin{cases} (\boldsymbol{P}^- \boldsymbol{\rho} - \boldsymbol{\rho}, \boldsymbol{v})_{D^\kappa} = 0, & \forall \boldsymbol{v} \in \boldsymbol{\mathcal{P}}^{k-1}(D^\kappa), \quad k \geq 1, \\ ((\boldsymbol{P}^- \boldsymbol{\rho} - \boldsymbol{\rho}) \cdot \boldsymbol{n}, w)_e = 0, & \forall w \in \mathcal{P}^k(e), \quad \forall e \in \partial D^\kappa, \quad e \neq \tilde{e}. \end{cases} \tag{4}$$

In this case, we represent the following lemmas, see [3].

Lemma 2 *Let* $\boldsymbol{P}$ *be the projection defined by* (3). *Then, for any* $\eta \in H^{k+1}(\Omega)$,

$$\|\boldsymbol{P}\eta - \eta\|_{L^2(\Omega_h)} + h^{1/2} \|\boldsymbol{P}\eta - \eta\|_{L^2(\xi_h)} \leq C h^{k+1} \|\eta\|_{H^{k+1}(\Omega_h)},$$

where C is independent of h. Let $\boldsymbol{P}^-$ *be the projection defined by* (4). *Then, for any* $\boldsymbol{\rho} \in [H^{k+1}(\Omega)]^d$,

$$\|\boldsymbol{P}^- \boldsymbol{\rho} - \boldsymbol{\rho}\|_{L^2(\Omega_h)} + h^{1/2} \|\boldsymbol{P}^- \boldsymbol{\rho} - \boldsymbol{\rho}\|_{L^2(\xi_h)} \leq C h^{k+1} \|\boldsymbol{\rho}\|_{H^{k+1}(\Omega_h)},$$

where C is independent of h.

In the sequel, we need the following inverse and trace inequalities.

Lemma 3 *For any* $\boldsymbol{v} \in \boldsymbol{\mathcal{P}}^k(D^\kappa)$ *and* $w \in \mathcal{P}^k(D^\kappa)$, *there exist positive constants C such that*

$$\begin{aligned} \|\boldsymbol{v} \cdot \boldsymbol{n}\|_{L^2(e)}^2 &\leq C h_{D^\kappa}^{-1} \|\boldsymbol{v}\|_{\boldsymbol{L}^2(D^\kappa)}^2, \\ \|w\|_{L^2(e)}^2 &\leq C h_{D^\kappa}^{-1} \|w\|_{L^2(D^\kappa)}^2, \\ \|\nabla w\|_{L^2(e)}^2 &\leq C h_{D^\kappa}^{-2} \|w\|_{L^2(D^\kappa)}^2, \end{aligned}$$

where e is a face of D^κ *and C is independent of the mesh size h.*

Lemma 4 *For any* $\boldsymbol{\rho} \in \boldsymbol{H}^1(D^\kappa)$ *and* $\eta \in H^1(D^\kappa)$, *there exist positive constants C such that*

$$\begin{aligned} \|\boldsymbol{\rho} \cdot \boldsymbol{n}\|_{L^2(e)}^2 &\leq C \|\boldsymbol{\rho}\|_{\boldsymbol{L}^2(D^\kappa)} \|\boldsymbol{\rho}\|_{\boldsymbol{H}^1(D^\kappa)}, \\ \|\eta\|_{L^2(e)}^2 &\leq C \|\eta\|_{L^2(D^\kappa)}^2 \|\eta\|_{H^1(D^\kappa)}^2, \end{aligned}$$

where e is a face of D^κ *and C is a constant independent of the mesh size h.*

3 Construction of the LDG Scheme

In this section, we construct a numerical scheme for solving problem (1). Let M be a positive integer, $\Delta t = T/M$ be the time step size, and $t_m = m\Delta t, m = 0, 1, \cdots, M$ denote the time mesh points. An approximation to a time-fractional derivative (2), called the L1 formula, can be obtained by a simple quadrature formula as [19, 23, 26]

$$\mathrm{D}_t^{\alpha} u(\cdot, t_n) = \frac{(\Delta t)^{1-\alpha}}{\mathbf{\Gamma}(2-\alpha)} \sum_{i=0}^{n-1} b_i \frac{u(\cdot, t_{n-i}) - u(\cdot, t_{n-i-1})}{\Delta t} + \gamma^n(\cdot),$$

where $b_i = (i+1)^{1-\alpha} - i^{1-\alpha}$ and γ^n is the truncation error with the bound

$$\|\gamma^n\|_{\infty} \leq C(\Delta t)^{2-\alpha}.$$

Here C is a constant which depends on u, α, and T. It is easy to check that $b_i > 0$ for each i, $1 = b_0 > b_1 > \cdots$, and $b_n \to 0$ as $n \to \infty$.

We rewrite (1) as a first-order system

$$\begin{cases} \mathrm{D}_t^{\alpha} u - \nabla \cdot \boldsymbol{q} = f(\mathrm{x}, t), \\ \boldsymbol{q} = \nabla u. \end{cases}$$

Let $u_h \in V_h^k$ and $\boldsymbol{q}_h = (p_h, q_h) \in \boldsymbol{V}_h^k$ be the approximations of $u(\cdot, t)$ and $\boldsymbol{q}(\cdot, t)$, respectively. We define a fully discrete LDG scheme as follows: find $(u_h, \boldsymbol{q}_h)$, such that for all test functions $(v, \boldsymbol{v})$ in the finite element space $V_h^k \times \boldsymbol{V}_h^k$,

$$\begin{cases} (u_h^m, v)_{D^{\kappa}} + \beta\left((\boldsymbol{q}_h^m, \nabla v)_{D^{\kappa}} - \langle \boldsymbol{n} \cdot \hat{\boldsymbol{q}}_h^m, v \rangle_{\partial D^{\kappa}}\right) \\ = \beta(f^m, v)_{D^{\kappa}} \\ + \sum_{i=1}^{m-1} (b_{i-1} - b_i)(u_h^{m-i}, v)_{D^{\kappa}} + b_{m-1}(u_h^0, v)_{D^{\kappa}}, \\ (\boldsymbol{q}_h^m, \boldsymbol{v})_{D^{\kappa}} + (u_h^m, \nabla \cdot \boldsymbol{v})_{D^{\kappa}} - \langle \hat{u}_h^m, \boldsymbol{n} \cdot \boldsymbol{v} \rangle_{\partial D^{\kappa}} = 0, \end{cases} \tag{5}$$

where $\beta = (\Delta t)^{\alpha} \mathbf{\Gamma}(2-\alpha)$. Equation (5) is obtained after integration by parts once. Here $\hat{u}_h$ and $\hat{\boldsymbol{q}}_h$ in (5) are the "numerical fluxes". To guarantee consistency, stability and optimal order of convergence, we must define these numerical fluxes carefully. The choice of the numerical fluxes is not unique and among several choices, for the Dirichlet boundary condition we adopt the central flux, defined as

$$\hat{u}_h^m = \frac{(u_h^m)^+ + (u_h^m)^-}{2}, \quad \hat{\boldsymbol{q}}_h^m = \frac{(\boldsymbol{q}_h^m)^+ + (\boldsymbol{q}_h^m)^-}{2}$$

at all internal edges, and at the external edges we use

$$\hat{u}_h^m = (u_h^m)^+ = (u_h^m)^-, \quad \hat{\boldsymbol{q}}_h^m = (\boldsymbol{q}_h^m)^+ = (\boldsymbol{q}_h^m)^-.$$

We recover the global fully discrete scheme (5) as

$$\begin{cases} (u_h^m, v) + \beta\left((\boldsymbol{q}_h^m, \nabla v) - \sum_{\kappa=1}^{K} \langle \boldsymbol{n} \cdot \hat{\boldsymbol{q}}_h^m, v \rangle_{\partial D^\kappa} \right) \\ = \beta(f^m, v) + \sum_{i=1}^{m-1} (b_{i-1} - b_i)(u_h^{m-i}, v) + b_{m-1}(u_h^0, v), \\ (\boldsymbol{q}_h^m, \boldsymbol{v}) + (u_h^m, \nabla \cdot \boldsymbol{v}) - \sum_{\kappa=1}^{K} \langle \hat{u}_h^m, \boldsymbol{n} \cdot \boldsymbol{v} \rangle_{\partial D^\kappa} = 0. \end{cases} \tag{6}$$

4 Stability Analysis and Error Estimates

To simplify the notations and without loss of generality, we consider the case $f \equiv 0$ in our analysis. We proceed to the numerical stability for the scheme (6). Before initiating the theoretical analysis, we need the following result [35].

Lemma 5 *If a mesh, consisting of K convex polygons D^κ, $\kappa = 1, \cdots, M$, is considered for Ω, then*

$$\sum_{\kappa=1}^{K} (\boldsymbol{n} \cdot \boldsymbol{u}, v)_{\partial D^\kappa} = \oint_{\Gamma} \{\boldsymbol{u}\} \cdot [v] \mathrm{d}s + \oint_{\Gamma_i} \{v\} \cdot [\boldsymbol{u}] \mathrm{d}s.$$

Theorem 1 (L^2-stability) *For homogeneous Dirichlet boundary conditions, the fully discrete LDG scheme (6) is unconditionally stable, and the numerical solution u_h^m satisfies*

$$\|u_h^m\| \le \|u_h^0\|, \quad m = 1, \cdots, M.$$

Proof Taking test functions $v = u_h^m, \boldsymbol{v} = \beta \boldsymbol{q}_h^m$ and summing all terms of (6), we have

$$\begin{aligned} &(u_h^m, u_h^m) + \beta(\boldsymbol{q}_h^m, \boldsymbol{q}_h^m) - \beta\left(\oint_{\Gamma} \hat{\boldsymbol{q}}_h \cdot \left[u_h\right] \mathrm{d}s + \oint_{\Gamma_i} \hat{u_h} \left[\boldsymbol{q}_h\right] \mathrm{d}s \right) \\ &+\beta \int_{\Omega} \nabla \cdot (\boldsymbol{q}_h^m u_h^m) \mathrm{d}x = \sum_{i=1}^{m-1} (b_{i-1} - b_i)(u_h^{m-i}, u_h^m) + b_{m-1}(u_h^0, u_h^m). \end{aligned} \tag{7}$$

Using integration by parts and Lemma 5, we get

$$\begin{aligned} \int_{\Omega} \nabla \cdot (\boldsymbol{q}_h u_h) \, \mathrm{d}x &= \sum_{\kappa=1}^{K} \int_{D^\kappa} \nabla \cdot (\boldsymbol{q}_h u_h) \mathrm{d}x = \sum_{\kappa=1}^{K} \oint_{\partial D^\kappa} \boldsymbol{n} \cdot (\boldsymbol{q}_h) u_h \mathrm{d}s \\ &= \oint_{\Gamma} \{\boldsymbol{q}_h\} \cdot \left[u_h\right] \mathrm{d}s + \oint_{\Gamma_i} \{u_h\} \left[\boldsymbol{q}_h\right] \mathrm{d}s \\ &= \oint_{\Gamma_i} (u_h^+ \boldsymbol{q}_h^+ \cdot \boldsymbol{n}^+ + u_h^- \boldsymbol{q}_h^- \cdot \boldsymbol{n}^-) \mathrm{d}s + \oint_{\Gamma_b} (u_h \boldsymbol{q}_h \cdot \boldsymbol{n}) \mathrm{d}s. \end{aligned} \tag{8}$$

With the central flux, we recover

$$\oint_{\Gamma} \hat{\boldsymbol{q}}_h \cdot [u_h] \mathrm{d}s + \oint_{\Gamma_i} \hat{u_h} [\boldsymbol{q}_h] \mathrm{d}s = \oint_{\Gamma_i} (u_h^+ \boldsymbol{q}_h^+ \cdot \boldsymbol{n}^+ + u_h^- \boldsymbol{q}_h^- \cdot \boldsymbol{n}^-) \mathrm{d}s + \oint_{\Gamma_b} (u_h \boldsymbol{q}_h \cdot \boldsymbol{n}) \mathrm{d}s. \quad (9)$$

Using relations (8) and (9), (7) can be written as

$$(u_h^m, u_h^m) + \beta(\boldsymbol{q}_h^m, \boldsymbol{q}_h^m) = \sum_{i=1}^{m-1} (b_{i-1} - b_i)(u_h^{m-i}, u_h^m) + b_{m-1}(u_h^0, u_h^m). \quad (10)$$

For $m = 1$, we can get

$$\|u_h^1\|^2 + \beta \|\boldsymbol{q}_h^1\|^2 = (u_h^0, u_h^1) \leq \frac{1}{2}\|u_h^0\|^2 + \frac{1}{2}\|u_h^1\|^2,$$

and immediately

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

Next, we suppose the following inequalities hold:

$$\|u_h^m\| \leq \|u_h^0\|, \qquad m = 1, \cdots, l.$$

With $m = l + 1$ in (10) and

$$\sum_{i=1}^{l} (b_{i-1} - b_i) + b_l = 1,$$

we have

$$\begin{aligned} \|u_h^{l+1}\|^2 + \beta \|\boldsymbol{q}_h^{l+1}\|^2 &= \sum_{i=1}^{l} (b_{i-1} - b_i)(u_h^{l+1-i}, u_h^{l+1}) + b_l(u_h^0, u_h^{l+1}) \\ &\leq \sum_{i=1}^{l} (b_{i-1} - b_i)\|u_h^{l+1-i}\| \|u_h^{l+1}\| + b_l \|u_h^0\| \|u_h^{l+1}\| \\ &\leq \sum_{i=1}^{l} (b_{i-1} - b_i)\|u_h^0\| \|u_h^{l+1}\| + b_l \|u_h^0\| \|u_h^{l+1}\| \\ &= \|u_h^0\| \|u_h^{l+1}\| \leq \tfrac{1}{2}\|u_h^0\|^2 + \tfrac{1}{2}\|u_h^{l+1}\|^2, \end{aligned}$$

then

$$\|u_h^{l+1}\| \leq \|u_h^0\|.$$

To examine the convergence of the scheme (6), we express the following result:

Theorem 2 *Let $u(\cdot, t_n)$ be the exact solution of problem (1) with homogeneous Dirichlet boundary conditions, which is sufficiently smooth with bounded derivatives, and u_h^n be the numerical solution of the fully discrete LDG scheme (6). There holds the following error estimate on Cartesian meshes*:

$$\|u(\cdot, t_n) - u_h^n\| \leq C(h^{k+1} + (\Delta t)^2 + (\Delta t)^{\frac{\alpha}{2}} h^{k+\frac{1}{2}} + (\Delta t)^{\frac{\alpha}{2}} h^{k+1}),$$

and on triangular meshes

$$\|u(\cdot,t_n)-u_h^n\| \le C(h^{k+1}+(\Delta t)^2+(\Delta t)^{\frac{\alpha}{2}}h^{k+\frac{1}{2}}+(\Delta t)^{\frac{\alpha}{2}}(h^k+h^{k+1})),$$

where C is a constant depending on T,α, and u.

Proof It is easy to verify that the exact solution to (6) satisfies

$$\begin{aligned}&(u(\mathrm{x},t_m),v)+\beta\big((\boldsymbol{q}(\mathrm{x},t_m),\nabla v)-\langle \boldsymbol{n}\cdot\boldsymbol{q}(\mathrm{x},t_m),v\rangle\big)+(\boldsymbol{q}(\mathrm{x},t_m),\boldsymbol{v})\\&+(u(\mathrm{x},t_m),\nabla\cdot\boldsymbol{v})-\langle u(\mathrm{x},t_m),\boldsymbol{n}\cdot\boldsymbol{v}\rangle-\sum_{i=1}^{m-1}(b_{i-1}-b_i)(u(\mathrm{x},t_{m-i}),v)\\&-b_{m-1}(u(\mathrm{x},t_0),v)+\beta(\gamma^m(\mathrm{x}),v)=0.\end{aligned}\tag{11}$$

Subtracting (6) from (11), we obtain the error equation

$$\begin{aligned}&(e_u^m,v)+\beta\Big((e_{\boldsymbol{q}}^m,\nabla v)-\langle\boldsymbol{n}\cdot\widehat{e_{\boldsymbol{q}}^m},v\rangle\Big)+(e_{\boldsymbol{q}}^m,\boldsymbol{v})+(e_u^m,\nabla\cdot\boldsymbol{v})-\langle\widehat{e_u^m},\boldsymbol{n}\cdot\boldsymbol{v}\rangle\\&-\sum_{i=1}^{m-1}(b_{i-1}-b_i)(e_u^{m-i},v)-b_{m-1}(e_u^0,v)+\beta(\gamma^m(\mathrm{x}),v)=0,\end{aligned}\tag{12}$$

where

$$e_u^m=u(\mathrm{x},t_m)-u_h^m,\qquad e_{\boldsymbol{q}}^m=\boldsymbol{q}(\mathrm{x},t_m)-\boldsymbol{q}_h^m.$$

We will use two projections $\Pi,\boldsymbol{P}$:

$$\begin{cases}e_u^m=u(\mathrm{x},t_m)-u_h^m=\boldsymbol{P}e_u^m-(\boldsymbol{P}u(\mathrm{x},t_m)-u(\mathrm{x},t_m)),\\ e_{\boldsymbol{q}}^m=\boldsymbol{q}(\mathrm{x},t_m)-\boldsymbol{q}_h^m=\Pi e_{\boldsymbol{q}}^m-(\Pi\boldsymbol{q}(\mathrm{x},t_m)-\boldsymbol{q}(\mathrm{x},t_m)).\end{cases}\tag{13}$$

Using (13), the error equation (12) can be written

$$\begin{aligned}&(\boldsymbol{P}e_u^m,v)+\beta\Big((\Pi e_{\boldsymbol{q}}^m,\nabla v)-\langle\boldsymbol{n}\cdot\widehat{\Pi e_{\boldsymbol{q}}^m},v\rangle\Big)+(\Pi e_{\boldsymbol{q}}^m,\boldsymbol{v})+(\boldsymbol{P}e_u^m,\nabla\cdot\boldsymbol{v})-\langle\widehat{\boldsymbol{P}e_u^m},\boldsymbol{n}\cdot\boldsymbol{v}\rangle\\&=b_{m-1}(\boldsymbol{P}e_u^0,v)+\textstyle\sum_{i=1}^{m-1}(b_{i-1}-b_i)(\boldsymbol{P}e_u^{m-i},v)\\&\quad-\beta(\gamma^m(\mathrm{x}),v)+(\boldsymbol{P}u(\mathrm{x},t_m)-u(\mathrm{x},t_m),v)+(\Pi\boldsymbol{q}(\mathrm{x},t_m)-\boldsymbol{q}(\mathrm{x},t_m),\boldsymbol{v})\\&\quad+\beta\Big((\Pi\boldsymbol{q}(\mathrm{x},t_m)-\boldsymbol{q}(\mathrm{x},t_m),\nabla v)-\langle\boldsymbol{n}\cdot(\widehat{\Pi\boldsymbol{q}}(\mathrm{x},t_m)-\boldsymbol{q}(\mathrm{x},t_m)),v\rangle\Big)\\&\quad+(\boldsymbol{P}u(\mathrm{x},t_m)-u(\mathrm{x},t_m),\nabla\cdot\boldsymbol{v})-\langle\widehat{\boldsymbol{P}u}(\mathrm{x},t_m)-u(\mathrm{x},t_m),\boldsymbol{n}\cdot\boldsymbol{v}\rangle\\&\quad-b_{m-1}(\boldsymbol{P}u(\mathrm{x},t_0)-u(\mathrm{x},t_0),v)\\&\quad-\sum_{i=1}^{m-1}(b_{i-1}-b_i)(\boldsymbol{P}u(\mathrm{x},t_{m-i})-u(\mathrm{x},t_{m-i}),v).\end{aligned}\tag{14}$$

In the following, we consider two cases.

4.1 Rectangular Mesh

Taking test functions $v=\boldsymbol{P}e_u^m$ and $\boldsymbol{v}=\beta\Pi e_{\boldsymbol{q}}^m$, for the homogeneous Dirichlet boundary condition, we obtain

$$\begin{aligned}\|\boldsymbol{P}e_u^m\|^2+\beta\|\Pi e_{\boldsymbol{q}}^m\|^2 &= b_{m-1}(\boldsymbol{P}e_u^0,\boldsymbol{P}e_u^m)+\sum_{i=1}^{m-1}(b_{i-1}-b_i)(\boldsymbol{P}e_u^{m-i},\boldsymbol{P}e_u^m)\\ &\quad-\beta(\gamma^m(\mathrm{x}),\boldsymbol{P}e_u^m)+(\boldsymbol{P}u(\mathrm{x},t_m)-u(\mathrm{x},t_m),\boldsymbol{P}e_u^m)\\ &\quad+\beta(\Pi\boldsymbol{q}(\mathrm{x},t_m)-\boldsymbol{q}(\mathrm{x},t_m),\Pi e_{\boldsymbol{q}}^m)-\beta\langle\widehat{\boldsymbol{P}u}(\mathrm{x},t_m)-u(\mathrm{x},t_m),\boldsymbol{n}\cdot\Pi e_{\boldsymbol{q}}^m\rangle\\ &\quad+\beta(\boldsymbol{P}u(\mathrm{x},t_m)-u(\mathrm{x},t_m),\nabla\cdot\Pi e_{\boldsymbol{q}}^m)-b_{m-1}(\boldsymbol{P}u(\mathrm{x},t_0)-u(\mathrm{x},t_0),\boldsymbol{P}e_u^m)\\ &\quad-\sum_{i=1}^{m-1}(b_{i-1}-b_i)(\boldsymbol{P}u(\mathrm{x},t_{m-i})-u(\mathrm{x},t_{m-i}),\boldsymbol{P}e_u^m).\end{aligned}$$

Using Lemma 1, we can write

$$\begin{aligned}&|-\langle\widehat{\boldsymbol{P}u}(\mathrm{x},t_m)-u(\mathrm{x},t_m),\boldsymbol{n}\cdot\Pi e_{\boldsymbol{q}}^m\rangle+(\boldsymbol{P}u(\mathrm{x},t_m)-u(\mathrm{x},t_m),\nabla\cdot\Pi e_{\boldsymbol{q}}^m)|\\ &\le C_1h^{k+1}\|\Pi e_{\boldsymbol{q}}^m\|,\end{aligned}\tag{15}$$

and then

$$\begin{aligned}\|\boldsymbol{P}e_u^m\|^2+\beta\|\Pi e_{\boldsymbol{q}}^m\|^2 &\le b_{m-1}(\boldsymbol{P}e_u^0,\boldsymbol{P}e_u^m)+\sum_{i=1}^{m-1}(b_{i-1}-b_i)(\boldsymbol{P}e_u^{m-i},\boldsymbol{P}e_u^m)\\ &\quad-\beta(\gamma^m(\mathrm{x}),\boldsymbol{P}e_u^m)+(\boldsymbol{P}u(\mathrm{x},t_m)-u(\mathrm{x},t_m),\boldsymbol{P}e_u^m)\\ &\quad+\beta(\Pi\boldsymbol{q}(\mathrm{x},t_m)-\boldsymbol{q}(\mathrm{x},t_m),\Pi e_{\boldsymbol{q}}^m)\\ &\quad+\beta C_1h^{k+1}\|\Pi e_{\boldsymbol{q}}^m\|-b_{m-1}(\boldsymbol{P}u(\mathrm{x},t_0)-u(\mathrm{x},t_0),\boldsymbol{P}e_u^m)\\ &\quad-\sum_{i=1}^{m-1}(b_{i-1}-b_i)(\boldsymbol{P}u(\mathrm{x},t_{m-i})-u(\mathrm{x},t_{m-i}),\boldsymbol{P}e_u^m).\end{aligned}\tag{16}$$

For $m=1$, we have

$$\begin{aligned}\|\boldsymbol{P}e_u^1\|^2+\beta\|\Pi e_{\boldsymbol{q}}^1\|^2 &\le(\boldsymbol{P}e_u^0,\boldsymbol{P}e_u^1)-\beta(\gamma^1(\mathrm{x}),\boldsymbol{P}e_u^1)\\ &\quad+(\boldsymbol{P}u(\mathrm{x},t_1)-u(\mathrm{x},t_1),\boldsymbol{P}e_u^1)+\beta(\Pi\boldsymbol{q}(\mathrm{x},t_1)-\boldsymbol{q}(\mathrm{x},t_1),\Pi e_{\boldsymbol{q}}^1)\\ &\quad+\beta C_1h^{k+1}\|\Pi e_{\boldsymbol{q}}^1\|-(\boldsymbol{P}u(\mathrm{x},t_0)-u(\mathrm{x},t_0),\boldsymbol{P}e_u^1),\end{aligned}$$

and hence

$$\begin{aligned}\|\boldsymbol{P}e_u^1\|^2+\beta\|\Pi e_{\boldsymbol{q}}^1\|^2 &\le\big(\|\boldsymbol{P}e_u^0\|+\beta\|\gamma^1(\mathrm{x})\|\\ &\quad+\|\boldsymbol{P}u(\mathrm{x},t_0)-u(\mathrm{x},t_0)\|+\|\boldsymbol{P}u(\mathrm{x},t_1)-u(\mathrm{x},t_1)\|\big)\times\|\boldsymbol{P}e_u^1\|\\ &\quad+\beta C_1h^{k+1}\|\Pi e_{\boldsymbol{q}}^1\|+\beta\|\Pi\boldsymbol{q}(\mathrm{x},t_1)-\boldsymbol{q}(\mathrm{x},t_1)\|\|\Pi e_{\boldsymbol{q}}^1\|.\end{aligned}$$

Then using the following facts:

$$\|P^-e_u^0\|\le Ch^{k+1},\quad ab\le\varepsilon a^2+\frac{1}{4\varepsilon}b^2,\tag{17}$$

$$\beta C_1h^{k+1}\|\Pi e_{\boldsymbol{q}}^1\|+\beta\|\Pi\boldsymbol{q}(\mathrm{x},t_1)-\boldsymbol{q}(\mathrm{x},t_1)\|\|\Pi e_{\boldsymbol{q}}^1\|\le C_1\beta h^{2k+2}+\beta\|\Pi e_{\boldsymbol{q}}^1\|^2,\tag{18}$$

and similar to the 1D cases [41, 42], we can write

$$\begin{aligned}\|\boldsymbol{P}e_u^1\|^2+\beta\|\Pi e_{\boldsymbol{q}}^1\|^2 &\le C_2(h^{k+1}+(\Delta t)^2+(\Delta t)^{\frac{\alpha}{2}}h^{k+\frac{1}{2}})^2\\ &\quad+\frac{1}{2}\|\boldsymbol{P}e_u^1\|^2+C_3\beta h^{2k+2}+\beta\|\Pi e_{\boldsymbol{q}}^1\|^2.\end{aligned}\tag{19}$$

Consequently,

$$\|\boldsymbol{P}e_u^1\|^2 \le C((h^{k+1}+(\Delta t)^2+(\Delta t)^{\frac{\alpha}{2}}h^{k+\frac{1}{2}})^2+(\Delta t)^{\alpha}h^{2(k+1)}).$$

Next we suppose the following inequalities hold:

$$\|\boldsymbol{P}e_u^m\| \le C(h^{k+1}+(\Delta t)^2+(\Delta t)^{\frac{\alpha}{2}}h^{k+\frac{1}{2}}+(\Delta t)^{\frac{\alpha}{2}}h^{k+1}), \quad m=1,2,\cdots,l.$$

For $m=l+1$, from (16), we can obtain

$$\begin{aligned}\|\boldsymbol{P}e_u^{l+1}\|^2+\beta\|\Pi e_{\boldsymbol{q}}^{l+1}\|^2 \le\ & b_l(\boldsymbol{P}e_u^0,\boldsymbol{P}e_u^{l+1})+\sum_{i=1}^{l}(b_{i-1}-b_i)(\boldsymbol{P}e_u^{l+1-i},\boldsymbol{P}e_u^{l+1})\\ &-\beta(\gamma^{l+1}(\mathrm{x}),\boldsymbol{P}e_u^{l+1})+(\boldsymbol{P}u(\mathrm{x},t_{l+1})-u(\mathrm{x},t_{l+1}),\boldsymbol{P}e_u^{l+1})\\ &+\beta(\Pi\boldsymbol{q}(\mathrm{x},t_{l+1})-\boldsymbol{q}(\mathrm{x},t_{l+1}),\Pi e_{\boldsymbol{q}}^{l+1})\\ &+\beta C_1h^{k+1}\|\Pi e_{\boldsymbol{q}}^{l+1}\|-b_l(\boldsymbol{P}u(\mathrm{x},t_0)-u(\mathrm{x},t_0),\boldsymbol{P}e_u^{l+1})\\ &-\sum_{i=1}^{l}(b_{i-1}-b_i)(\boldsymbol{P}u(\mathrm{x},t_{l+1-i})-u(\mathrm{x},t_{l+1-i}),\boldsymbol{P}e_u^{l+1})\end{aligned}$$

and hence

$$\begin{aligned}\|\boldsymbol{P}e_u^{l+1}\|^2+\beta\|\Pi e_{\boldsymbol{q}}^{l+1}\|^2 \le\ & \Bigg(b_l\|\boldsymbol{P}e_u^0\|+\sum_{i=1}^{l}(b_{i-1}-b_i)\|\boldsymbol{P}e_u^{l+1-i}\|+\beta\|\gamma^m(\mathrm{x})\|\\ &+b_l\|\boldsymbol{P}u(\mathrm{x},t_0)-u(\mathrm{x},t_0)\|+\|\boldsymbol{P}u(\mathrm{x},t_{l+1})-u(\mathrm{x},t_{l+1})\|\\ &+\sum_{i=1}^{l}(b_{i-1}-b_i)\|\boldsymbol{P}u(\mathrm{x},t_{l+1-i})-u(\mathrm{x},t_{l+1-i})\|\Bigg)\times\|\boldsymbol{P}e_u^{l+1}\|\\ &+\beta C_1h^{k+1}\|\Pi e_{\boldsymbol{q}}^{l+1}\|+\beta\|\Pi\boldsymbol{q}(\mathrm{x},t_{l+1})-\boldsymbol{q}(\mathrm{x},t_{l+1})\|\|\Pi e_{\boldsymbol{q}}^{l+1}\|.\end{aligned}$$

Then using facts (17), (18), and $\sum_{i=1}^{l}(b_{i-1}-b_i)+b_l=1$, we can write

$$\begin{aligned}\|\boldsymbol{P}e_u^{l+1}\|^2+\beta\|\Pi e_{\boldsymbol{q}}^{l+1}\|^2 \le\ & C_2(h^{k+1}+(\Delta t)^2+(\Delta t)^{\frac{\alpha}{2}}h^{k+\frac{1}{2}})^2\\ &+\frac{1}{2}\|\boldsymbol{P}e_u^{l+1}\|^2+C_3\beta h^{2k+2}+\beta\|\Pi e_{\boldsymbol{q}}^{l+1}\|^2.\end{aligned}$$

Consequently,

$$\|\boldsymbol{P}e_u^{l+1}\|^2 \le C((h^{k+1}+(\Delta t)^2+(\Delta t)^{\frac{\alpha}{2}}h^{k+\frac{1}{2}})^2+(\Delta t)^{\alpha}h^{2(k+1)}).$$

4.2 Triangular Mesh

Taking test functions $v=\boldsymbol{P}e_u^m$ and $\boldsymbol{v}=\beta\Pi e_{\boldsymbol{q}}^m$, for the Dirichlet boundary condition, we obtain

$$\begin{aligned}\|\boldsymbol{P}e_u^m\|^2+\beta\|\Pi e_{\boldsymbol{q}}^m\|^2 &= b_{m-1}(\boldsymbol{P}e_u^0,\boldsymbol{P}e_u^m)+\sum_{i=1}^{m-1}(b_{i-1}-b_i)(\boldsymbol{P}e_u^{m-i},\boldsymbol{P}e_u^m)\\ &\quad-\beta(\gamma^m(\mathrm{x}),\boldsymbol{P}e_u^m)+(\boldsymbol{P}u(\mathrm{x},t_m)-u(\mathrm{x},t_m),\boldsymbol{P}e_u^m)\\ &\quad+\beta(\Pi\boldsymbol{q}(\mathrm{x},t_m)-\boldsymbol{q}(\mathrm{x},t_m),\Pi e_{\boldsymbol{q}}^m)-\beta\langle\widehat{\boldsymbol{P}u}(\mathrm{x},t_m)-u(\mathrm{x},t_m),\boldsymbol{n}\cdot\Pi e_{\boldsymbol{q}}^m\rangle\\ &\quad-b_{m-1}(\boldsymbol{P}u(\mathrm{x},t_0)-u(\mathrm{x},t_0),\boldsymbol{P}e_u^m)\\ &\quad-\sum_{i=1}^{m-1}(b_{i-1}-b_i)(\boldsymbol{P}u(\mathrm{x},t_{m-i})-u(\mathrm{x},t_{m-i}),\boldsymbol{P}e_u^m).\end{aligned}\tag{20}$$

When $m=1$, (20) is

$$\begin{aligned}\|\boldsymbol{P}e_u^1\|^2+\beta\|\Pi e_{\boldsymbol{q}}^1\|^2 &= (\boldsymbol{P}e_u^0,\boldsymbol{P}e_u^1)-\beta(\gamma^1(\mathrm{x}),\boldsymbol{P}e_u^1)\\ &\quad+(\boldsymbol{P}u(\mathrm{x},t_1)-u(\mathrm{x},t_1),\boldsymbol{P}e_u^1)+\beta(\Pi\boldsymbol{q}(\mathrm{x},t_1)-\boldsymbol{q}(\mathrm{x},t_1),\Pi e_{\boldsymbol{q}}^1)\\ &\quad-\beta\langle\widehat{\boldsymbol{P}u}(\mathrm{x},t_1)-u(\mathrm{x},t_1),\boldsymbol{n}\cdot\Pi e_{\boldsymbol{q}}^1\rangle-(\boldsymbol{P}u(\mathrm{x},t_0)-u(\mathrm{x},t_0),\boldsymbol{P}e_u^1).\end{aligned}\tag{21}$$

Using the inverse and trace inequalities and Lemmas 3 and 4, we get

$$\begin{aligned}&-\beta\langle\widehat{\boldsymbol{P}u}(\mathrm{x},t_1)-u(\mathrm{x},t_1),\boldsymbol{n}\cdot\Pi e_{\boldsymbol{q}}^1\rangle\\ &\le C_1\beta\|\widehat{\boldsymbol{P}u}(\mathrm{x},t_1)-u(\mathrm{x},t_1)\|_{L^2(\partial\Omega_h)}\|\boldsymbol{n}\cdot\Pi e_{\boldsymbol{q}}^1\|_{L^2(\partial\Omega_h)}\\ &\le C_2\beta h^k\|\Pi e_{\boldsymbol{q}}^1\|\le C_3\beta h^{2k}+\frac{1}{2}\beta\|\Pi e_{\boldsymbol{q}}^1\|^2.\end{aligned}\tag{22}$$

Using (22), (21) can be written as

$$\begin{aligned}\|\boldsymbol{P}e_u^1\|^2+\beta\|\Pi e_{\boldsymbol{q}}^1\|^2 &\le \|\big(\boldsymbol{P}e_u^0\|+\beta\|\gamma^1(\mathrm{x})\|+\|\boldsymbol{P}u(\mathrm{x},t_0)-u(\mathrm{x},t_0)\|\\ &\quad+\|\boldsymbol{P}u(\mathrm{x},t_1)-u(\mathrm{x},t_1)\|\big)\times\|\boldsymbol{P}e_u^1\|\\ &\quad+C_3\beta h^{2k}+\frac{1}{2}\beta\|\Pi e_{\boldsymbol{q}}^1\|^2+\beta\|\Pi\boldsymbol{q}(\mathrm{x},t_1)-\boldsymbol{q}(\mathrm{x},t_1)\|\|\Pi e_{\boldsymbol{q}}^1\|.\end{aligned}$$

Then using (17)–(19), we obtain

$$\begin{aligned}\|\boldsymbol{P}e_u^1\|^2+\beta\|\Pi e_{\boldsymbol{q}}^1\|^2 &\le C_4(h^{k+1}+(\Delta t)^2+(\Delta t)^{\frac{\alpha}{2}}h^{k+\frac{1}{2}})^2+\frac{1}{2}\|\boldsymbol{P}e_u^1\|^2\\ &\quad+C_5\beta h^{2k+2}+C_3\beta h^{2k}+\beta\|\Pi e_{\boldsymbol{q}}^1\|^2.\end{aligned}$$

Consequently, since $\beta=(\Delta t)^\alpha\boldsymbol{\Gamma}(2-\alpha)$, using a big enough C, we have

$$\|\boldsymbol{P}e_u^1\|^2\le C((h^{k+1}+(\Delta t)^2+(\Delta t)^{\frac{\alpha}{2}}h^{k+\frac{1}{2}})^2+(\Delta t)^\alpha h^{2k}+(\Delta t)^\alpha h^{2(k+1)}).$$

Next, we suppose the following inequalities hold:

$$\|\boldsymbol{P}e_u^m\|\le C(h^{k+1}+(\Delta t)^2+(\Delta t)^{\frac{\alpha}{2}}h^{k+\frac{1}{2}}+(\Delta t)^{\frac{\alpha}{2}}(h^k+h^{k+1})),\quad m=1,2,\cdots,l.$$

It can be derived similar relation for $m=l+1$.

More precisely, there holds the following error estimate on Cartesian meshes:

$$\|u(\cdot,t_n)-u_h^n\|\le C(h^{k+1}+(\Delta t)^2+(\Delta t)^{\frac{\alpha}{2}}h^{k+\frac{1}{2}}),$$

and on triangular meshes

$$\|u(\cdot, t_n) - u_h^n\| \leq C(h^{k+1} + (\Delta t)^2 + (\Delta t)^{\frac{\alpha}{2}} h^k).$$

For small time step sizes, we can get optimal order of convergence.

5 Numerical Examples

In this section, we perform numerical experiments of the LDG method applied to the time fractional diffusion equation. We check the spatial accuracy by fixing the time step sufficiently small to avoid the contamination of the temporal error. We have verified that the results shown are numerically convergent in all cases with the aid of successive mesh refinements.

Example 1 We consider (1) with the exact solution $u(\mathrm{x}, t) = t^2 \sin(x_1) \sin(x_2)$ on $\Omega = (0, \pi) \times (0, \pi)$. Obviously, we encounter with the homogeneous Dirichlet boundary conditions, and

$$f(\mathrm{x}, t) = \left(2t^2 + \frac{2t^{2-\alpha}}{\Gamma(3-\alpha)}\right) \sin(x_1) \sin(x_2).$$

We take piecewise P^2 polynomials as the basis functions and investigate the accuracy of the proposed method. Setting $T = 1$ and Δt very small and using the usual L^2 and L^∞ error norms, we prepare results of Table 1 for Cartesian meshes and results of Table 2 for triangular meshes for several values of α. The order of convergence (OC) of the method is evidently about 3. The errors in L^2-norm and L^∞-norm for piecewise $P^k, k = 1, 2, 3$ polynomials for $\alpha = 0.1, 0.9$ are presented in Fig. 1 for triangular meshes and in Fig. 2 for Cartesian meshes. Now, we consider $\Omega = (0, 2\pi) \times (0, 2\pi)$ and, therefore, we face periodic boundary conditions. We repeat the previous tests and report results in Table 3 for Cartesian meshes

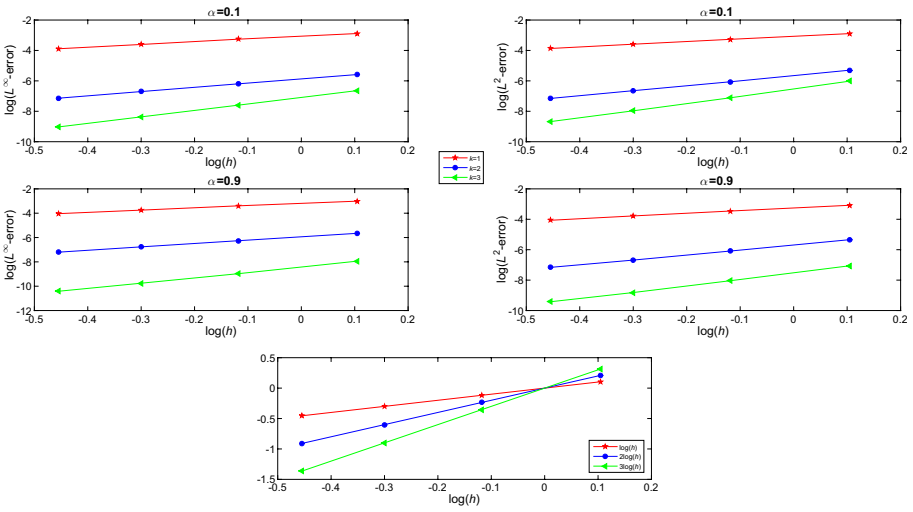

Fig. 1 Natural log(error) as a function of natural log(h) for $\alpha = 0.1, 0.9$ when using piecewise $P^k, k = 1, 2, 3$ polynomials with triangular meshes. The lowest picture is for reference lines

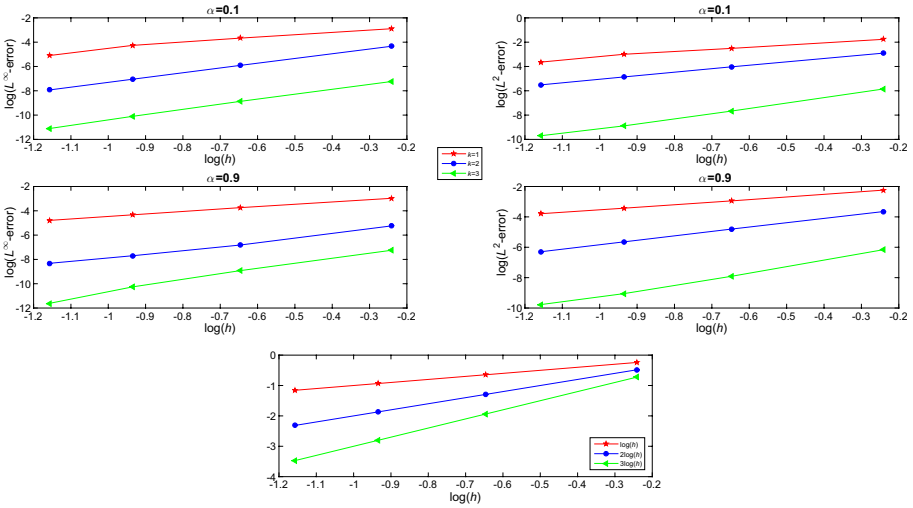

Fig. 2 Natural log(error) as a function of natural log(h) for $\alpha = 0.1, 0.9$ when using piecewise $P^k, k = 1, 2, 3$ polynomials for Cartesian meshes. The lowest picture is for reference lines

and in Table 4 for triangular meshes. The OC of the method is evidently about 3. The errors in L^2-norm and L^∞-norm for piecewise $P^k, k = 1, 2, 3$ polynomials for $\alpha = 0.1, 0.9$ are presented in Fig. 3 for triangular meshes and in Fig. 4 for Cartesian meshes. To interpret better, we report data related to Figs. 1, 2, 3, 4 in Tables 5, 6, 7, and 8 and find that by choosing small time step sizes for both Cartesian and triangular meshes and for both homogeneous Dirichlet boundary conditions and periodic boundary conditions, the OC with respect to the spatial variable converges to $k + 1$.

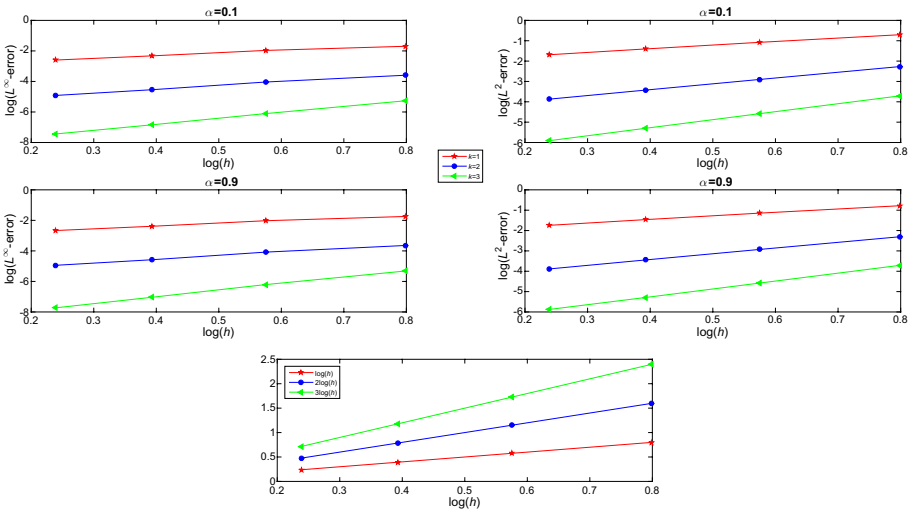

Fig. 3 Natural log(error) as a function of natural log(h) for $\alpha = 0.1, 0.9$ when using piecewise $P^k, k = 1, 2, 3$ polynomials for triangular meshes. The lowest picture is for reference lines

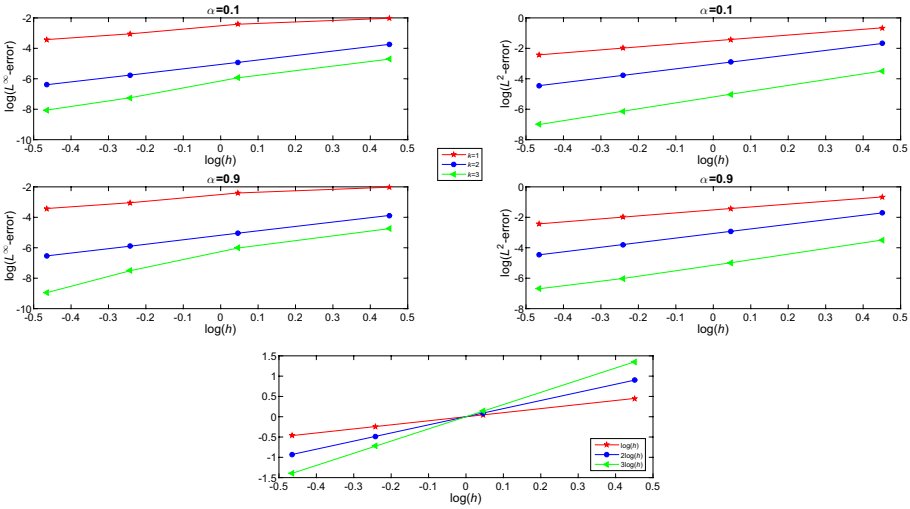

Fig. 4 Natural log(error) as a function of natural log(h) for $\alpha = 0.1, 0.9$ when using piecewise $P^k, k = 1, 2, 3$ polynomials for Cartesian meshes. The lowest picture is for reference lines

Table 1 Accuracy test for Example 1 for Cartesian meshes

	K	L^∞-error	OC	L^2-error	OC
$\alpha = 0.1$	16	0.013 074	–	0.054 876	–
	36	0.002 705	3.89	0.017 823	2.77
	64	0.000 869	3.95	0.007 758	2.89
	100	0.000 359	3.96	0.004 029	2.94
$\alpha = 0.3$	16	0.010 647	–	0.045 654	–
	36	0.002 201	3.89	0.014 774	2.78
	64	0.000 707	3.95	0.006 422	2.90
	100	0.000 306	3.75	0.003 333	2.94
$\alpha = 0.5$	16	0.008 538	–	0.037 762	–
	36	0.001 764	3.89	0.012 157	2.80
	64	0.000 566	3.95	0.005 273	3.90
	100	0.000 278	3.19	0.002 733	2.94
$\alpha = 0.7$	16	0.006 750	–	0.031 221	–
	36	0.001 394	3.89	0.009 976	2.81
	64	0.000 487	3.66	0.004 313	2.91
	100	0.000 252	2.95	0.002 232	2.95
$\alpha = 0.9$	16	0.005 285	–	0.025 980	–
	36	0.001 094	3.88	0.008 217	2.84
	64	0.000 454	3.06	0.003 542	2.92
	100	0.000 243	2.80	0.001 842	2.93

Table 2 Accuracy test for Example 1 for triangular meshes

	K	L^∞-error	OC	L^2-error	OC
$\alpha = 0.1$	32	0.003 764	–	0.005 012	–
	50	0.002 040	2.75	0.002 338	3.42
	72	0.001 226	2.79	0.001 299	3.22
	98	0.000 794	2.82	0.000 782	3.29
$\alpha = 0.3$	32	0.003 708	–	0.004 968	–
	50	0.002 011	2.74	0.002 358	3.34
	72	0.001 210	2.78	0.001 287	3.31
	98	0.000 784	2.82	0.000 774	3.30
$\alpha = 0.5$	32	0.003 648	–	0.004 921	–
	50	0.001 981	2.74	0.002 335	3.34
	72	0.001 192	2.79	0.001 274	3.32
	98	0.000 773	2.81	0.000 766	3.30
$\alpha = 0.7$	32	0.003 583	–	0.004 867	–
	50	0.001 948	2.73	0.002 307	3.35
	72	0.001 173	2.78	0.001 257	3.33
	98	0.000 731	3.07	0.000 755	3.31
$\alpha = 0.9$	32	0.003 508	–	0.004 767	–
	50	0.001 908	2.73	0.002 274	3.25
	72	0.001 150	2.78	0.001 251	3.28
	98	0.000 745	2.82	0.000 777	3.09

Table 3 Accuracy test for Example 1 for Cartesian meshes

	K	L^∞-error	OC	L^2-error	OC
$\alpha = 0.1$	16	0.023 846	–	0.186 717 039	–
	36	0.007 266	2.93	0.054 766 859	3.02
	64	0.003 164	2.89	0.022 902 703	3.03
	100	0.001 681	2.84	0.011 673 543	3.02
$\alpha = 0.3$	16	0.023 152	–	0.185 419 389	–
	36	0.007 089	2.92	0.054 544 273	3.02
	64	0.003 105	2.87	0.022 842 346	3.03
	100	0.001 656	2.82	0.011 652 186	3.02
$\alpha = 0.5$	16	0.022 419	–	0.184 078 220	–
	36	0.006 902	2.91	0.054 315 078	3.01
	64	0.003 040	2.85	0.022 778 925	3.02
	100	0.001 627	2.80	0.011 628 691	3.01
$\alpha = 0.7$	16	0.021 632	–	0.182 687 996	–
	36	0.006 699	2.89	0.054 067 171	3.00
	64	0.002 950	2.85	0.022 702 178	3.02
	100	0.001 583	2.79	0.011 594 720	3.01
$\alpha = 0.9$	16	0.020 731	–	0.181 183 153	–
	36	0.006 453	2.88	0.053 736 153	3.00
	64	0.002 755	2.86	0.022 563 026	3.02
	100	0.001 455	2.86	0.011 516 954	3.01

Table 4 Accuracy test for Example 1 for triangular meshes

	K	L^∞-error	OC	L^2-error	OC
$\alpha = 0.1$	32	0.017 642	–	0.054 930	–
	50	0.010 644	2.77	0.032 583	2.87
	72	0.007 318	2.43	0.020 852	2.90
	98	0.004 977	2.98	0.014 125	2.92
$\alpha = 0.3$	32	0.017 488	–	0.054 617	–
	50	0.010 565	2.76	0.032 440	2.86
	72	0.007 284	2.41	0.020 782	2.89
	98	0.004 963	2.87	0.014 087	2.91
$\alpha = 0.5$	32	0.017 323	–	0.054 281	–
	50	0.010 481	2.76	0.032 290	2.83
	72	0.007 247	2.39	0.020 708	2.88
	98	0.004 946	2.86	0.014 047	2.91
$\alpha = 0.7$	32	0.017 144	–	0.053 928	–
	50	0.010 387	2.75	0.032 130	2.84
	72	0.007 202	2.38	0.020 626	2.88
	98	0.004 924	2.85	0.014 000	2.90
$\alpha = 0.9$	32	0.016 914	–	0.053 524	–
	50	0.010 251	2.75	0.031 930	2.83
	72	0.007 122	2.36	0.020 513	2.87
	98	0.004 872	2.84	0.013 930	2.90

Table 5 Some results related to Fig. 1

		K	L^∞-error	OC	L^2-error	OC
$k = 1$	$\alpha = 0.1$	32	0.055 784	–	0.055 465	–
		50	0.038 440	1.67	0.038 057	1.69
		72	0.027 128	1.91	0.027 601	1.76
		98	0.020 490	1.82	0.020 862	1.82
	$\alpha = 0.9$	32	0.049 285	–	0.046 032	–
		50	0.033 552	1.72	0.031 530	1.70
		72	0.023 551	1.94	0.022 808	1.78
		98	0.017 666	1.87	0.017 168	1.84
$k = 3$	$\alpha = 0.1$	32	0.003 110	–	0.002 439	–
		50	0.000 502	4.30	0.000 816	4.91
		72	0.000 232	4.23	0.000 346	4.71
		98	0.000 120	4.28	0.000 170	4.58
	$\alpha = 0.9$	32	0.000 357	–	0.000 863	–
		50	0.000 127	4.63	0.000 323	4.40
		72	0.000 058	4.30	0.000 148	4.28
		98	0.000 030	4.28	0.000 081	3.91

Table 6 Some results related to Fig. 2

		K	L^∞-error	OC	L^2-error	OC
$k=1$	$\alpha=0.1$	16	0.055 330	–	0.170 899	–
		36	0.025 534	1.91	0.081 244	1.83
		64	0.014 057	2.06	0.050 047	1.68
		100	0.006 095	3.74	0.026 218	2.90
	$\alpha=0.9$	16	0.051 186	–	0.106 494	–
		36	0.023 734	1.90	0.052 513	1.74
		64	0.013 155	2.05	0.032 480	1.67
		100	0.008 311	2.06	0.022 559	1.63
$k=3$	$\alpha=0.1$	16	0.000 719	–	0.002 869	–
		36	0.000 140	4.04	0.000 469	4.47
		64	0.000 041	4.27	0.000 138	4.24
		100	0.000 015	4.51	0.000 061	3.65
	$\alpha=0.9$	16	0.000 711	–	0.002 107	–
		36	0.000 133	4.13	0.000 367	4.31
		64	0.000 035	4.64	0.000 117	3.97
		100	0.000 009	6.09	0.000 050	3.79

Table 7 Some results related to Fig. 3

		K	L^∞-error	OC	L^2-error	OC
$k=1$	$\alpha=0.1$	32	0.183 192	–	0.493 864	–
		50	0.139 907	1.21	0.340 334	1.67
		72	0.097 790	1.96	0.245 848	1.78
		98	0.074 373	1.78	0.184 873	1.84
	$\alpha=0.9$	32	0.174 722	–	0.457 642	–
		50	0.132 235	1.25	0.318 609	1.62
		72	0.091 834	2.00	0.231 622	1.75
		98	0.069 533	1.80	0.174 861	1.82
$k=3$	$\alpha=0.1$	32	0.005 151	–	0.024 455	–
		50	0.002 231	3.75	0.010 236	3.90
		72	0.001 067	4.05	0.004 988	3.94
		98	0.000 586	3.89	0.002 708	3.96
	$\alpha=0.9$	32	0.004 901	–	0.024 433	–
		50	0.002 017	3.98	0.010 267	3.89
		72	0.000 877	4.57	0.005 049	3.89
		98	0.000 439	4.49	0.002 804	3.82

Table 8 Some results related to Fig. 4

		K	L^∞-error	OC	L^2-error	OC
$k=1$	$\alpha=0.1$	16	0.132 275	–	0.521 210	–
		36	0.089 292	0.97	0.241 716	1.90
		64	0.047 482	2.20	0.137 419	1.96
		100	0.032 602	1.68	0.088 298	1.98
	$\alpha=0.9$	16	0.132 626	–	0.516 207	–
		36	0.090 588	0.94	0.241 042	1.88
		64	0.047 724	2.23	0.137 113	1.96
		100	0.032 654	1.70	0.088 030	1.99
$k=3$	$\alpha=0.1$	16	0.009 026	–	0.030 639	–
		36	0.002 644	3.03	0.006 637	3.77
		64	0.000 714	4.55	0.002 173	3.88
		100	0.000 318	3.62	0.000 908	3.91
	$\alpha=0.9$	16	0.008 722	–	0.030 579	–
		36	0.002 436	3.14	0.006 825	3.70
		64	0.000 549	5.18	0.002 430	3.59
		100	0.000 130	6.46	0.001 240	3.01

6 Conclusion

In this paper, we have developed an LDG method for 2D time fractional diffusion equations. The numerical stability and convergence of the method for both rectangular and triangular meshes have been theoretically proven. The numerical results demonstrate the applicability and efficiency of the proposed method.

Compliance with Ethical Standards

Conflict of interest There is no conflict of interest.

Appendix A

Another approximations to the time-fractional derivative (2) are L1-2 and L1-2-3 formulae [16, 32] which can be obtained by using quadratic and cubic interpolation formulae, respectively. The order of convergence with respect to the time variable for L1, L1-2, and L1-2-3 formulas are $2-\alpha$, $3-\alpha$, and $4-\alpha$, respectively. We follow here just L1-2 formula which is

$$\mathrm{D}_t^\alpha u(\cdot,t_n)=\frac{(\Delta t)^{1-\alpha}}{\Gamma(2-\alpha)}\sum_{i=0}^{n-1}c_i\frac{u(\cdot,t_{n-i})-u(\cdot,t_{n-i-1})}{\Delta t}+\gamma^n(\cdot),$$

where $c_0=1$ for $n=1$; and for $n\geq 2$,

$$c_j=\begin{cases}a_0+d_0, & j=0,\\ a_j+d_j-d_{j-1}, & 1\leq j\leq n-2,\\ a_j-d_{j-1}, & j=n-1,\end{cases}$$

where

$$a_j = (j+1)^{(1-\alpha)} - j^{(1-\alpha)}; \quad 0 \le j \le n-1,$$

and

$$d_j = [(j+1)^{(2-\alpha)} - j^{(2-\alpha)}]/(2-\alpha) - [(j+1)^{(1-\alpha)} + j^{(1-\alpha)}]/2; \quad j \ge 0,$$

and γ^n is the truncation error with the estimate

$$\|\gamma^n\| \le \begin{cases} C(\Delta t)^{2-\alpha}, & n = 1, \\ C(\Delta t)^{3-\alpha}, & n \ge 2. \end{cases}$$

Then, we can define a fully discrete LDG scheme as follows: find $(u_h, \boldsymbol{q}_h)$, such that for all test functions $(v, \boldsymbol{v}) \in V_h^k \times \boldsymbol{V}_h^k$,

$$\begin{cases} (u_h^m, v) + \beta\left((\boldsymbol{q}_h^m, \nabla v) - \sum_{\kappa=1}^{K} \langle \boldsymbol{n} \cdot \hat{\boldsymbol{q}}_h^m, v \rangle_{\partial D^\kappa} \right) \\ = \beta(f^m, v) + \sum_{i=1}^{m-1} (c_{i-1} - c_i)(u_h^{m-i}, v) + c_{m-1}(u_h^0, v), \\ (\boldsymbol{q}_h^m, \boldsymbol{v}) + (u_h^m, \nabla \cdot \boldsymbol{v}) - \sum_{\kappa=1}^{K} \langle \hat{u}_h^m, \boldsymbol{n} \cdot \boldsymbol{v} \rangle_{\partial D^\kappa} = 0, \end{cases} \tag{A1}$$

where $\beta = (\Delta t)^\alpha \boldsymbol{\Gamma}(2-\alpha)$. In order to examine the convergence of the scheme (A1), we express the following result.

Theorem A1 *Let $u(\cdot, t_n)$ be the exact solution of problem (1) with homogeneous Dirichlet boundary conditions, which is sufficiently smooth with bounded derivatives, and u_h^n be the numerical solution of the LDG scheme (23). There holds the following error estimate on Cartesian meshes*:

$$\|u(\cdot, t_n) - u_h^n\| \le \begin{cases} C(h^{k+1} + (\Delta t)^2 + (\Delta t)^{\frac{\alpha}{2}} h^{k+\frac{1}{2}}), & n = 1, \\ C(h^{k+1} + (\Delta t)^2 + (\Delta t)^{\frac{\alpha}{2}} h^{k+\frac{1}{2}}), & n \ge 2, \end{cases}$$

and on triangular meshes

$$\|u(\cdot, t_n) - u_h^n\| \le \begin{cases} C(h^{k+1} + (\Delta t)^2 + (\Delta t)^{\frac{\alpha}{2}} h^k), & n = 1, \\ C(h^{k+1} + (\Delta t)^2 + (\Delta t)^{\frac{\alpha}{2}} h^k), & n \ge 2, \end{cases}$$

where C is a constant depending on T, α, and u.

Proof It is more or less similar to the proof of Theorem 2.

References

1. Basu, T.S., Wang, H.: A fast second-order finite difference method for space-fractional diffusion equations. Int. J. Numer. Anal. Model. **9**, 658–666 (2012)
2. Carella, A.R., Dorao, C.A.: Least-squares spectral method for the solution of a fractional advection–dispersion equation. J. Comput. Phys. **232**, 33–45 (2013)
3. Cockburn, B., Dong, B.: An analysis of the minimal dissipation local discontinuous Galerkin method for convection–diffusion problems. J. Sci. Comput. **32**, 233–262 (2007)
4. Cockburn, B., Kanschat, G., Perugia, I., Schotzau, D.: Superconvergence of the local discontinuous Galerkin method for elliptic problems on Cartesian grids. SIAM J. Numer. Anal. **39**, 264–285 (2002)
5. Cockburn, B., Shu, C.W.: The local discontinuous Galerkin method for time-dependent convection–diffusion systems. SIAM J. Numer. Anal. **35**, 2440–2463 (1998)
6. Cui, M.R.: Compact finite difference method for the fractional diffusion equation. J. Comput. Phys. **228**, 7792–7804 (2009)
7. Deng, W.H., Hesthaven, J.S.: Local discontinuous Galerkin methods for fractional diffusion equations. Math. Model. Numer. Anal. **47**, 1845–1864 (2013)
8. Deng, W.H., Hesthaven, J.S.: Local discontinuous Galerkin methods for fractional ordinary differential equations. BIT Numer. Math. **55**, 967–985 (2015)
9. Ding, H.F., Li, C.P.: Mixed spline function method for reaction–subdiffusion equation. J. Comput. Phys. **242**, 103–123 (2013)
10. Dong, B., Shu, C.W.: Analysis of a local discontinuous Galerkin method for linear time-dependent fourth-order problems. SIAM J. Numer. Anal. **47**, 3240–3268 (2009)
11. Du, R., Cao, W.R., Sun, Z.Z.: A compact difference scheme for the fractional diffusion–wave equation. Appl. Math. Model. **34**, 2998–3007 (2010)
12. Ervin, V.J., Heuer, N., Roop, J.P.: Numerical approximation of a time dependent, nonlinear, space-fractional diffusion equation. SIAM J. Numer. Anal. **45**, 572–591 (2007)
13. Eshaghi, J., Kazem, S., Adibi, H.: The local discontinuous Galerkin method for 2D nonlinear time-fractional advection–diffusion equations. Eng. Comput. **35**, 1317–1332 (2019). https://doi.org/10.1007/s00366-018-0665-8
14. Fix, G., Roop, J.: Least squares finite element solution of a fractional order two-point boundary value problem. Comput. Math. Appl. **48**, 1017–1033 (2004)
15. Gao, G.H., Sun, Z.Z.: A compact finite difference scheme for the fractional sub-diffusion equations. J. Comput. Phys. **230**, 586–595 (2011)
16. Gao, G.H., Sun, Z., Zhang, H.: A new fractional numerical differentiation formula to approximate the Caputo fractional derivative and its applications. J. Comput. Phys. **259**, 33–50 (2014)
17. He, J.H., Wu, X.H.: Variational iteration method: new development and applications. Comput. Math. Appl. **54**, 881–894 (2007)
18. Huang, C., An, N., Yu, X.: A fully discrete direct discontinuous Galerkin method for the fractional diffusion–wave equation. Appl. Anal. **97**, 659–675 (2018)
19. Jiang, Y., Ma, J.: High-order finite element methods for time-fractional partial differential equations. J. Comput. Appl. Math. **235**, 3285–3290 (2011)
20. Jin, B., Lazarov, R., Liu, Y., Zhou, Z.: The Galerkin finite element method for a multi-term time-fractional diffusion equation. J. Comput. Phys. **281**, 825–843 (2015)
21. Karaa, S., Mustapha, K., Pani, Amiya K.: Finite volume element method for two-dimensional fractional subdiffusion problems. IMA J. Numer. Anal. **37**, 945–964 (2017)
22. Kochubei, A., Luchko, Y. (eds.) Handbook of Fractional Calculus with Applications, vol. 1. Basic Theory. Walter de Gruyter GmbH, Berlin (2019)
23. Li, C.Z., Chen, Y.: Numerical approximation of nonlinear fractional differential equations with subdiffusion and superdiffusion. Comput. Math. Appl. **62**, 855–875 (2011)
24. Li, X.J., Xu, C.J.: A space–time spectral method for the time fractional diffusion equation. SIAM J. Numer. Anal. **47**, 2108–2131 (2009)
25. Li, C.P., Zeng, F.H.: The finite difference methods for fractional ordinary differential equations. Numer. Funct. Anal. Optim. **34**, 149–179 (2013)
26. Lin, Y.M., Xu, C.J.: Finite difference/spectral approximations for the time-fractional diffusion equation. J. Comput. Phys. **225**, 1533–1552 (2007)
27. Liu, Y., Shu, C.W., Zhang, M.: Superconvergence of energy-conserving discontinuous Galerkin methods for linear hyperbolic equations. Commun. Appl. Math. Comput. **1**, 101–116 (2019)
28. Liu, F., Zhuang, P., Burrage, K.: Numerical methods and analysis for a class of fractional advection–dispersion models. Comput. Math. Appl. **64**, 2990–3007 (2012)

29. Meerschaert, M.M., Tadjeran, C.: Finite difference approximations for fractional advection–dispersion. J. Comput. Appl. Math. **172**, 65–77 (2004)
30. Metzler, R., Klafter, J.: The random walk's guide to anomalous diffusion: a fractional dynamics approach. Phys. Rep. **339**, 1–77 (2000)
31. Miller, K.S., Ross, B.: An Introduction to the Fractional Calculus and Fractional Differential Equations. Wiley, New York (1993)
32. Mokhtari, R., Mostajeran, F.: A high order formula to approximate the Caputo fractional derivative. Commun. Appl. Math. Comput. **2**, 1–29 (2020)
33. Oldham, K., Spanier, J.: The Fractional Calculus: Theory and Applications of Differentiation and Integration of Arbitray Order. Academic Press, New York (1974)
34. Podlubny, I., Chechkin, A., Skovranek, T., Chen, Y.Q., Jara, B.M.V.: Matrix approach to discrete fractional calculus II: partial fractional differential equations. J. Comput. Phys. **228**, 3137–3153 (2009)
35. Qiu, L., Deng, W., Hesthaven, J.S.: Nodal discontinuous Galerkin methods for fractional diffusion equations on 2D domain with triangular meshes. J. Comput. Phys. **298**, 678–694 (2015)
36. Roop, J.P.: Computational aspects of FEM approximation of fractional advection dispersion equations on bounded domains in $\mathbb{R}^2$. J. Comput. Appl. Math. **193**, 243–268 (2006)
37. Wang, H., Zhang, Q., Wang, S., Shu, C.W.: Local discontinuous Galerkin methods with explicit-implicit-null time discretizations for solving nonlinear diffusion problems. Sci. China Math. **63**(1), 183–204 (2020). https://doi.org/10.1007/s11425-018-9524-x
38. Wang, H., Zhang, Q., Shu, C.W.: Third order implicit-explicit Runge–Kutta local discontinuous Galerkin methods with suitable boundary treatment for convection–diffusion problems with Dirichlet boundary conditions. J. Comput. Appl. Math. **342**, 164–179 (2018)
39. Xu, Q., Zheng, Z.: Discontinuous Galerkin method for time fractional diffusion equation. J. Inf. Comput. Sci. **11**, 3253–3264 (2013)
40. Yang, Q.Q., Turner, I., Liu, F., Ilic, M.: Novel numerical methods for solving the time space fractional diffusion equation in two dimensions. SIAM J. Sci. Comput. **33**, 1159–1180 (2011)
41. Yeganeh, S., Mokhtari, R., Fouladi, S.: Using an LDG method for solving an inverse source problem of the time-fractional diffusion equation. Iran. J. Numer. Anal. Optim. **7**, 115–135 (2017)
42. Yeganeh, S., Mokhtari, R., Hesthaven, J.S.: Space-dependent source determination in a time-fractional diffusion equation using a local discontinuous Galerkin method. BIT Numer. Math. **57**, 685–707 (2017)
43. Zhang, X., Tang, B., He, Y.: Homotopy analysis method for higher-order fractional integro-differential equations. Comput. Math. Appl. **62**, 3194–3203 (2011)
44. Zhao, Y., Chen, P., Bu, W., Liu, X., Tang, Y.: Two mixed finite element methods for time-fractional diffusion equations. J. Sci. Comput. **70**, 407–428 (2017)
45. Zhuang, P., Liu, F.: Finite difference approximation for two-dimensional time fractional diffusion equation. J. Algorithm Comput. Tech. **1**, 1–15 (2007)