fractal and fractional

MDPI

Article

Numerical Simulation Based on Interpolation Technique for Multi-Term Time-Fractional Convection–Diffusion Equations

Xindong Zhang [1], Yan Chen [2,*], Leilei Wei [3] and Sunil Kumar [4]

[1] College of Big Data Statistics, Guizhou University of Finance and Economics, Guiyang 550025, China; liaoyuan1126@163.com
[2] College of General Studies, Xinjiang University of Technology, Hotan 848000, China
[3] School of Mathematics and Statistics, Henan University of Technology, Zhengzhou 450001, China
[4] Department of Mathematics, National Institute of Technology, Jamshedpur 831014, Jharkhand, India
* Correspondence: cyan19981022@163.com; Tel.: +86-1887-903-0746

Abstract: This paper introduces a novel approach for solving multi-term time-fractional convection–diffusion equations with the fractional derivatives in the Caputo sense. The proposed highly accurate numerical algorithm is based on the barycentric rational interpolation collocation method (BRICM) in conjunction with the Gauss–Legendre quadrature rule. The discrete scheme constructed in this paper can achieve high computational accuracy with very few interval partitioning points. To verify the effectiveness of the present discrete scheme, some numerical examples are presented and are compared with the other existing method. Numerical results demonstrate the effectiveness of the method and the correctness of the theoretical analysis.

Keywords: Caputo derivative; barycentric rational interpolation; multi-term time-fractional convection–diffusion equation; Gauss–Legendre quadrature rule

1. Introduction

In this study, we investigate the following multi-term time-fractional convection–diffusion equations with the fractional derivatives in the Caputo sense,

$$\begin{cases} \left({}_0^C D_t^{\beta} + \sum_{j=1}^{J} \lambda_j \, {}_0^C D_t^{\beta_j}\right) u(x,t) = P(x,t) u_{xx}(x,t) - Q(x,t) u_x(x,t) + f(x,t), & \\ & (x,t) \in \Omega, \\ u(x,0) = \psi(x), & x \in (0,L), \\ u(0,t) = u(L,t) = 0, & t \in (0,T], \end{cases} \tag{1}$$

where $0 < \beta_s \leq \cdots \leq \beta_2 \leq \beta_1 < \beta < 1$ are the fractional orders, $\lambda_j > 0$ $(1 \leq j \leq J)$, and $P(x,t)$ and $Q(x,t)$ are the diffusion coefficient and the convection coefficient, respectively. $\Omega = [0,L] \times [0,T]$, $f(x,t)$ is the forcing function, $\psi(x)$ is given sufficiently smooth function, and $u(x,t)$ is the unknown function.

If $Q(x,t) = 0$, then Equation (1) will become the multi-term time-fractional diffusion equation (TFDE). And if $J = 0$, then Equation (1) will become the time-fractional convection–diffusion equation, which has been studied by many scholars using various numerical methods, including the finite difference method [1–4], the finite element method [5–7], the finite volume method [8–10] and the spectral method [11,12], etc.

Multi-term fractional order differential equations provide a higher degree of flexibility in modelling complex real-world phenomena. They can be used to capture a wider range of memory effects by combining multiple fractional orders, which makes them more effective in modelling complex systems. Therefore, it is necessary to explore the numerical method of multi-term fractional order differential equations. A number of studies on multi-term

check for updates

Citation: Zhang, X.; Chen, Y.; Wei, L.; Kumar, S. Numerical Simulation Based on Interpolation Technique for Multi-Term Time-Fractional Convection–Diffusion Equations. *Fractal Fract.* **2024**, *8*, 687. https://doi.org/10.3390/fractalfract8120687

Received: 15 October 2024
Revised: 9 November 2024
Accepted: 20 November 2024
Published: 23 November 2024

fractional order differential equations have been conducted recently, in particular, studies such as those of the Hermite wavelets approach [13], the Pseudospectral method [14], Chebyshev polynomials [15,16], the generalized squared remainder minimization method [17], the Haar wavelet collocation method [18], and so on.

The main purpose of this paper is to solve a class of multi-term fractional convection–diffusion equations using the BRICM, where the fractional order derivatives are in turn given by the Caputo definition:

$$ {}_0^C D_t^{\beta} u(x,t) = \frac{1}{\Gamma(1-\beta)} \int_0^t (t-s)^{-\beta} \frac{\partial u(x,s)}{\partial s} \mathrm{d}s, \tag{2} $$

which is one of the common derivatives of fractional order and has been applied in many areas. Properties and more details about Caputo's fractional derivative can be found in [19–21]. The BRICM is a high-order interpolation algorithm, which can effectively avoid the Runge's phenomenon and has good robustness to irregular data. In recent years, the BRICM has been applied in solving various differential equations. Additional studies can be found in [22–27], among others.

The rest of this paper is organized as follows: In Section 2, the discrete scheme is constructed by using the combination of the BRICM and the Gauss–Legendre quadrature rule, the theoretical analysis is given in Section 3, and the numerical results in Section 4 support the theoretical analysis. Finally, we conclude our results in Section 5.

2. Highly Accurate Numerical Algorithm for Equation (1)

2.1. Background Knowledge of the BRICM

For classical rational interpolation, the existence of poles has a significant negative impact on it. Therefore, Berrut and Mittelmann [28] proposed an interpolation technique to avoid poles and improved the result by using higher-order rational functions. The interpolation can be written in the following barycentric rational form:

$$ u(x) \approx \sum_{i=0}^{I} \frac{w_i}{x-x_i} u_i \Big/ \sum_{p=0}^{I} \frac{w_p}{x-x_p}, \tag{3} $$

where x_i ($i \in [I]$, $[I] = \{0, 1, \cdots, I\}$) are $I+1$ different interpolation nodes, the value of $u(x)$ at point x_i is denoted by $u_i = u(x_i)$. In [29], Berrut used

$$ w_i = (-1)^i, \ i \in [I] \tag{4} $$

to denote the interpolation weights of barycentric rational interpolation. Let d_x $(0 \leq d_x \leq I)$ be an arbitrary integer; in [30], Floater and Hormann used the interpolation weights as

$$ w_i = \sum_{z \in [\mathcal{O}_i]} (-1)^z \prod_{\theta=z, \theta \neq i}^{z+d_x} \frac{1}{x_i - x_\theta}, \ [\mathcal{O}_i] = \{z \in [I] : i - d_x \leq z \leq i\}, \tag{5} $$

and if $d_x = 0$, we can get that $w_i = (-1)^i$ $(i \in [I])$ which is the same as that in [29]. Thus, in the following, we just focus on the case of $d_x \geq 1$.

Let $\phi_i(x) = \frac{w_i}{x-x_i} / \sum_{p=0}^{I} \frac{w_p}{x-x_p}$, then $\phi_i(x) \to 1$ as $x \to x_i$, and $\phi_i(x) = 0$ as $x \neq x_i$. By Equation (3), we can obtain the barycentric rational interpolation function (BRIF) of $u(x)$, denoted by $u_{B_1}(x)$ as

$$ u(x) \approx u_{B_1}(x) = \sum_{i=0}^{I} \phi_i(x) u_i. \tag{6} $$

Similarly, by Equation (6), the transcription in the time domain implies that

$$ u(t) \approx u_{B_1}(t) = \sum_{k=0}^{K} \vartheta_k(t) u_k, \tag{7} $$

where $u_k = u(t_k)$ and $\vartheta_k(t) = \frac{\omega_k}{t-t_k} / \sum_{\rho=0}^{K} \frac{\omega_\rho}{t-t_\rho}$.

Let $u_i(t) = u(x_i, t)$ and $u_{ik} = u(x_i, t_k)$. By Equation (7), the BRIF of $u_i(t)$ is denoted by $u_{B_1}(x_i, t)$, then we have

$$u_i(t) \approx u_{B_1}(x_i, t) = \sum_{k=0}^{K} \vartheta_k(t) u_{ik}. \tag{8}$$

Next, we will consider the BRIF of $u(x,t)$ at interpolation nodes (x_i, t_k) $(i \in [I],\ k \in [K])$. Similar to Equation (6), we can get the following BRIF of $u(x,t)$ denoted by $u_{B_2}(x,t)$,

$$u(x,t) \approx \sum_{i=0}^{I} \phi_i(x) u_i(t) \approx \sum_{i=0}^{I} \phi_i(x) u_{B_1}(x_i, t) = \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(x) \vartheta_k(t) u_{ik} = u_{B_2}(x,t). \tag{9}$$

In this paper, the second class of Chebyshev nodes ($x_i = \cos \frac{i\pi}{I}, i = 0, 1, \cdots, I$) will be used for analysis and calculation.

2.2. The Differential Matrices

In this subsection, we will consider the differential matrix of barycentric rational interpolation. As in [31], we can obtain the BRIF for the μ-order ($\mu \in N^+$) derivative of $u(x,t)$ on nodes (x_n, t_m) $(n \in [I],\ m \in [K])$

$$\begin{aligned} \frac{\partial^{(\mu)} u_{B_2}}{\partial x^\mu} &= \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i^{(\mu)}(x_n) \vartheta_k(t_m) u_{ik} = \sum_{i=0}^{I} \sum_{k=0}^{K} A_{ni}^{(\mu)} \vartheta_k(t_m) u_{ik}, \\ \frac{\partial^{(\mu)} u_{B_2}}{\partial t^\mu} &= \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(x_n) \vartheta_k^{(\mu)}(t_m) u_{ik} = \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(x_n) B_{mk}^{(\mu)} u_{ik}, \end{aligned} \tag{10}$$

where $A_{ni}^{(\mu)}$ and $B_{mk}^{(\mu)}$ are the μ-order differential matrices for the corresponding variables, respectively. As in [31], the form of $A_{ni}^{(\mu)}$ and $B_{mk}^{(\mu)}$ can be obtained as follows,

$$A_{ni}^{(1)} = \begin{cases} \frac{w_i}{w_n} \frac{1}{x_n - x_i}, & n \neq i, \\ -\sum\limits_{l=0, l\neq n}^{n} A_{nl}^{(1)}, & n = i, \end{cases} \quad A_{ni}^{(\mu)} = \begin{cases} \mu \left(A_{ni}^{(1)} A_{nn}^{(\mu-1)} - \frac{A_{ni}^{(\mu-1)}}{x_n - x_i} \right), & n \neq i, \\ -\sum\limits_{l=0, l\neq n}^{I} A_{nl}^{(\mu)}, & n = i, \end{cases}$$

$$B_{mk}^{(1)} = \begin{cases} \frac{w_k}{w_m} \frac{1}{t_m - t_k}, & m \neq k, \\ -\sum\limits_{q=0, q\neq m}^{K} B_{mq}^{(1)}, & m = k, \end{cases} \quad B_{mk}^{(\mu)} = \begin{cases} \mu \left(B_{mk}^{(1)} B_{mm}^{(\mu-1)} - \frac{B_{mk}^{(\mu-1)}}{t_m - t_k} \right), & m \neq k, \\ -\sum\limits_{q=0, q\neq m}^{K} B_{mq}^{(\mu)}, & m = k. \end{cases}$$

2.3. Approximate Scheme of the Caputo Derivative

We consider the approximate scheme of the Caputo derivative in this subsection. By Equation (2), we can infer that

$${}_0^C D_t^\beta u(x,t) = \frac{1}{\Gamma(2-\beta)} \frac{\partial u(x,0)}{\partial s} t^{1-\beta} + \frac{1}{\Gamma(2-\beta)} \int_0^t (t-s)^{1-\beta} \frac{\partial^2 u(x,s)}{\partial s^2} \mathrm{d}s. \tag{11}$$

In Equation (11), by using Equation (10) and discretizing the domain by $I+1$ $(0 = x_0 < x_1 < \cdots < x_I = 1)$ nodes in space and $K+1$ $(0 = t_0 < t_1 < \cdots < t_K = 1)$ nodes in time, a preliminary discrete scheme for the Caputo derivative can be obtained as

$$\begin{aligned} &{}_0^C \mathcal{D}_t^\beta u(x,t) \\ &= \frac{t^{1-\beta}}{\Gamma(2-\beta)} \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(x) \vartheta_k'(t_0) u_{ik} + \frac{1}{\Gamma(2-\beta)} \int_0^t (t-s)^{1-\beta} \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(x) \vartheta_k''(s) u_{ik} \mathrm{d}s. \end{aligned} \tag{12}$$

For the second term at the right-hand side of Equation (12), by using the Gauss–Legendre quadrature rule, that is $\int_0^t f(s)ds \approx \sum_{r=1}^{R} W_r f(s_r)$ (see [32] for more details), we can get the discrete scheme of it. Denote

$$\begin{aligned}&{}_0^C\mathfrak{D}_t^{\beta} u(x,t)\\&= \frac{1}{\Gamma(2-\beta)} \sum_{i=0}^{I} \sum_{k=0}^{K} \Big(t^{1-\beta}\phi_i(x)\vartheta_k'(t_0)u_{ik} + \sum_{r=1}^{R} (t-s_r)^{1-\beta}\phi_i(x)\vartheta_k''(s_r)W_r u_{ik}\Big).\end{aligned} \tag{13}$$

Then, we can obtain the fully discrete scheme of the Caputo derivative as ${}_0^C D_t^{\beta} u(x,t)$, that is ${}_0^C D_t^{\beta} u(x,t) \approx {}_0^C\mathfrak{D}_t^{\beta} u(x,t)$.

2.4. Discrete Scheme of Equation (1)

In this subsection, the fully discrete scheme of Equation (1) will be given based on the BRICM. Applying Equations (10) and (13), we get

$$\begin{aligned}&\frac{1}{\Gamma(2-\beta)} \Big(t^{1-\beta} \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(x)\vartheta_k'(t_0)u_{ik} + \sum_{i=0}^{I} \sum_{k=0}^{K} \sum_{r=1}^{R} (t-s_r)^{1-\beta}\phi_i(x)\vartheta_k''(s_r)W_r u_{ik}\Big)\\&+ \sum_{j=1}^{J} \lambda_j \frac{1}{\Gamma(2-\beta_j)} \sum_{i=0}^{I} \sum_{k=0}^{K} \Big(t^{1-\beta_j}\phi_i(x)\vartheta_k'(t_0)u_{ik} + \sum_{r=1}^{R} (t-s_r)^{1-\beta_j}\phi_i(x)\vartheta_k''(s_r)W_r u_{ik}\Big)\\&= \sum_{i=0}^{I} \sum_{k=0}^{K} \Big(P(x,t)\phi_i''(x) - Q(x,t)\phi_i'(x)\Big)\vartheta_k(t)u_{ik} + f(x,t).\end{aligned} \tag{14}$$

Let $L_k(t_m) = \sum_{r=1}^{R} (t_m - s_r)^{1-\beta}\vartheta_k''(s_r)W_r$ and $L_{jk}(t_m) = \sum_{r=1}^{R} (t_m - s_r)^{1-\beta_j}\vartheta_k''(s_r)W_r$, combining these results, ($\vartheta_k'(t_0) = B_{0k}^{(1)}$, $\phi_i(x_n) = \delta_{in}$, $\phi_i'(x_n) = A_{ni}^{(1)}$ and $\phi_i''(x_n) = A_{ni}^{(2)}$), and by Equation (14), we can get

$$\begin{aligned}&\frac{1}{\Gamma(2-\beta)} \Big(t_m^{1-\beta} \sum_{i=0}^{I} \sum_{k=0}^{K} \delta_{in}B_{0k}^{(1)}u_{ik} + \sum_{i=0}^{I} \sum_{k=0}^{K} \delta_{in}L_k(t_m)u_{ik}\Big)\\&+ \sum_{j=1}^{J} \lambda_j \frac{1}{\Gamma(2-\beta_j)} \sum_{i=0}^{I} \sum_{k=0}^{K} \Big(t_m^{1-\beta_j}\delta_{in}B_{0k}^{(1)}u_{ik} + \delta_{in}L_{jk}(t_m)u_{ik}\Big)\\&= \sum_{i=0}^{I} \sum_{k=0}^{K} \Big(P(x_n,t_m)A_{ni}^{(2)} - Q(x_n,t_m)A_{ni}^{(1)}\Big)\delta_{km}u_{ik} + f(x_n,t_m).\end{aligned} \tag{15}$$

Taking all values of $n \in [I]$ and $m \in [K]$, the fully discrete scheme of Equation (1) can be expressed as

$$\begin{aligned}&\Big[\frac{1}{\Gamma(2-\beta)}\Big(\boldsymbol{I}_N \otimes \boldsymbol{B}^{(1)} + \boldsymbol{I}_N \otimes \boldsymbol{L}\Big) + \sum_{j=1}^{J} \lambda_j \frac{1}{\Gamma(2-\beta_j)}\Big(\boldsymbol{I}_N \otimes \boldsymbol{B}_j^{(1)} + \boldsymbol{I}_N \otimes \boldsymbol{L}_j\Big)\\&-\boldsymbol{P}(\boldsymbol{A}^{(2)} \otimes \boldsymbol{I}_M) - \boldsymbol{Q}(\boldsymbol{A}^{(1)} \otimes \boldsymbol{I}_M)\Big]\boldsymbol{U} = \boldsymbol{F},\end{aligned} \tag{16}$$

where $\boldsymbol{U} = [\boldsymbol{u}_0 \cdots \boldsymbol{u}_I]'$ with $\boldsymbol{u}_i = [u_{i0} \cdots u_{iK}]'$, $\boldsymbol{F} = [\boldsymbol{f}_0 \cdots \boldsymbol{f}_I]'$ with $\boldsymbol{f}_i = [f(x_i,t_0) \cdots f(x_i,t_K)]'$,

$$\boldsymbol{B}^{(1)} = \begin{bmatrix} t_0^{1-\beta}B_{00}^{(1)} & \cdots & t_0^{1-\beta}B_{0K}^{(1)} \\ \vdots & \ddots & \vdots \\ t_K^{1-\beta}B_{00}^{(1)} & \cdots & t_K^{1-\beta}B_{0K}^{(1)} \end{bmatrix}, \quad \boldsymbol{L} = \begin{bmatrix} L_0(t_0) & \cdots & L_K(t_0) \\ \vdots & \ddots & \vdots \\ L_0(t_K) & \cdots & L_K(t_K) \end{bmatrix},$$

$$\boldsymbol{B}_j^{(1)} = \begin{bmatrix} t_0^{1-\beta_j}B_{00}^{(1)} & \cdots & t_0^{1-\beta_j}B_{0K}^{(1)} \\ \vdots & \ddots & \vdots \\ t_K^{1-\beta_j}B_{00}^{(1)} & \cdots & t_K^{1-\beta_j}B_{0K}^{(1)} \end{bmatrix}, \quad \boldsymbol{L}_j = \begin{bmatrix} L_{j0}(t_0) & \cdots & L_{jK}(t_0) \\ \vdots & \ddots & \vdots \\ L_{j0}(t_K) & \cdots & L_{jK}(t_K) \end{bmatrix},$$

$$\boldsymbol{A}^{(\mu)} = \begin{bmatrix} A_{00}^{(\mu)} & \cdots & A_{0I}^{(\mu)} \\ \vdots & \ddots & \vdots \\ A_{I0}^{(\mu)} & \cdots & A_{II}^{(\mu)} \end{bmatrix} \text{ with } \mu = 1, 2,$$

$\boldsymbol{P} = diag\{P_0, P_1, \cdots, P_I\}$ with $P_i = [P(x_i, t_0) \cdots P(x_i, t_K)]'$, and $\boldsymbol{Q} = diag\{Q_0, Q_1, \cdots, Q_I\}$ with $Q_i = [Q(x_i, t_0) \cdots Q(x_i, t_K)]'$. $\boldsymbol{I}_M$ and $\boldsymbol{I}_N$ are the identity matrices of order $K+1$ and $I+1$, respectively, and the discrete formats of the initial value conditions are

$$u_{B_2}(x, 0) = \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(x)\vartheta_k(0)u_{ik} = \psi(x),$$
$$u_{B_2}(0, t) = \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(0)\vartheta_k(t)u_{ik} = 0 \text{ and } u_{B_2}(L, t) = \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(L)\vartheta_k(t)u_{ik} = 0.$$

3. Convergence Analysis

Set $h = \max\{h_i\}$ and $h_i = |x_{i+1} - x_i|$ with $i \in [I-1]$, and set $\tau = \max\{\tau_k\}$ and $\tau_k = |t_{k+1} - t_k|$ with $k \in [K-1]$. Let $u_i = u(x_i)$, $u_k = u(t_k)$ and $u_{ik} = u(x_i, t_k)$. Let $\mathcal{J}_1$ be the function space consisting of the interpolated basis functions $\{\phi_i\}_{i=0}^{I}$ defined by Equation (6), and $\mathcal{J}_2$ be the function space consisting of $\{\vartheta_k\}_{k=0}^{K}$ defined by Equation (7). We first give the following definitions and lemmas, which will be used in the following discussion.

Definition 1. *Suppose $u(x) \in C[0,1]$ and $u(t) \in C[0,1]$. Let $\mathfrak{R}_{x,I}$: $C[0,1] \to \mathcal{J}_1$ and $\mathfrak{R}_{t,K}$: $C[0,1] \to \mathcal{J}_2$ be the interpolation operators for x and t, respectively. They still satisfy the requirement that*

$$\mathfrak{R}_{x,I}u(x) = \sum_{i=0}^{I} \phi_i(x)u_i, \;\; \mathfrak{R}_{t,K}u(t) = \sum_{k=0}^{K} \vartheta_k(t)u_k.$$

Similarly, let $\mathcal{J} = \mathcal{J}_1 \bigcup \mathcal{J}_2$; we can define $\mathfrak{R}_{x,I}\mathfrak{R}_{t,K}$: $C([0,1] \times [0,1]) \to \mathcal{J}$, and it satisfies

$$\mathfrak{R}_{x,I}\mathfrak{R}_{t,K}u(x,t) = \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(x)\vartheta_k(t)u_{ik}.$$

It is obvious that $u_{B_2} := \mathfrak{R}_{x,I}\mathfrak{R}_{t,K}u$, and $\mathfrak{R}_{x,I}$, $\mathfrak{R}_{t,K}$, $\mathfrak{R}_{x,I}\mathfrak{R}_{t,K}$ are linear operators.

Definition 2 ([33]). *(Lebesgue constant)* $\boldsymbol{\lambda}_{I,d_x} = ||\mathfrak{R}_{x,I}||_\infty = \max\limits_{x\in[0,1]} \sum_{i=0}^{I} |\phi_i(x)|$, $\boldsymbol{\lambda}_{K,d_t} = ||\mathfrak{R}_{t,K}||_\infty = \max\limits_{t\in[0,1]} \sum_{k=0}^{K} |\vartheta_k(t)|$.

Lemma 1 ([34]). *For any set of well-spaced interpolation nodes, any μ_1 with $0 \le \mu_1 \le d_x$, and $u(x) \in C^{d_x+2+\mu_1}[0,1]$, then*

$$|u^{(\mu_1)}(x) - u_{B_1}^{(\mu_1)}(x)| \le C_1 h^{d_x+1-\mu_1}$$

and more specifically, $|u^{(\mu_1)}(x) - u_{B_1}^{(\mu_1)}(x)| \le C_1 h_i^{d_x+1-\mu_1}$, where C_1 is a constant, u_{B_1} is the BRIF of $u(x)$, $x \in [x_i, x_{i+1}]$ and $i \in [I-1]$.

Lemma 2 ([35]). *When the BRICM at quasi-equidistant nodes $(x_0, \cdots, x_I) \in \mathbb{R}^{(I+1)}$ with the basis function of Equation (6), its Lebesgue constant $\boldsymbol{\lambda}_{I,d_x}$ satisfies*

$$\boldsymbol{\lambda}_{I,d_x} \le (2 + H_x \ln I) * \begin{cases} \frac{3}{4}H_x, & if\ d_x = 0, \\ 2^{d_x-1}H_x^{d_x}, & if\ d_x \ge 1, \end{cases}$$

where $H_x \ge h/h_x$, $h_x = \min\{|x_{i+1} - x_i|\}$ and $i \in [I-1]$.

Lemma 3 ([32]). *Let $u^{(2R)}(t) \in C[0,T]$; there exists $\gamma \in (0,T)$, and the error estimate of the Gauss–Legendre quadrature rule can be presented as follows*

$$\int_0^T u(s)ds - \sum_{r=1}^{R} W_r u(s_r) = \frac{T^{2R+1}(R!)^4}{(2R+1)[(2R)!]^3} u^{(2R)}(\gamma),$$

where s_r and W_r are the integral points and integral weights of the Gauss–Legendre quadrature rule, respectively, and R is the number of points.

By a similar analysis, the following theorems can be obtained, as performed in [34,35].

Theorem 1. *Let $u = u(x,t)$ be the exact solution of Equation (1) and $u_{B_2} = u_{B_2}(x,t)$ be the numerical solution of Equation (16), and let $u_x^{(d_x+2)}(x,t)$, $u_t^{(d_t+2)}(x,t) \in C(\Omega)$ with $d_x \geq 1$ and $d_t \geq 1$, then*

$$||u - u_{B_2}||_\infty \leq C_1 h^{d_x+1} + C_2(2 + H_x ln\ I)2^{d_x-1} H_x^{d_x} \tau^{d_t+1}$$

where C_1, C_2 are all constants.

Proof. Applying triangle inequality, we can assert that

$$\begin{aligned} ||u - u_{B_2}||_\infty &= ||u - \mathfrak{R}_{x,I}\mathfrak{R}_{t,K}u||_\infty = ||u - \mathfrak{R}_{x,I}u + \mathfrak{R}_{x,I}u - \mathfrak{R}_{x,I}\mathfrak{R}_{t,K}u||_\infty \\ &\leq ||u - \mathfrak{R}_{x,I}u||_\infty + ||\mathfrak{R}_{x,I}u - \mathfrak{R}_{x,I}\mathfrak{R}_{t,K}u||_\infty. \end{aligned}$$

From Lemma 1, it follows that

$$||u - \mathfrak{R}_{x,I}u||_\infty \leq C_1 h^{d_x+1} \text{ and } ||u - \mathfrak{R}_{t,K}u||_\infty \leq C_2 \tau^{d_t+1}. \tag{17}$$

Since $\mathfrak{R}_{x,I}$ and $\mathfrak{R}_{t,K}$ are linear operators, Definition 2 and Equation (17) show that

$$||\mathfrak{R}_{x,I}u - \mathfrak{R}_{x,n}\mathfrak{R}_{t,K}u||_\infty \leq ||\mathfrak{R}_{x,I}(u - \mathfrak{R}_{t,K}u)||_\infty \leq ||\mathfrak{R}_{x,I}||_\infty ||u - \mathfrak{R}_{t,K}u||_\infty \leq C_2 \lambda_{I,d_x} \tau^{d_t+1}.$$

According to the above remark and Lemma 2, we have

$$||\mathfrak{R}_{x,K}u - \mathfrak{R}_{x,I}\mathfrak{R}_{t,K}u||_\infty \leq C_2(2 + H_x ln\ I)2^{d_x-1} H_x^{d_x} \tau^{d_t+1}. \tag{18}$$

By Equation (17) and Equation (18), we conclude that

$$||u - u_{B_2}||_\infty \leq C_1 h^{d_x+1} + C_2(2 + H_x ln\ I)2^{d_x-1} H_x^{d_x} \tau^{d_t+1}.$$

This completes the proof. □

Theorem 2. *Let ${}_0^C\mathfrak{D}_t^\beta u(x,t)$ be the fully discrete scheme of ${}_0^C D_t^\beta u(x,t)$ as in Equation (13) and let $u_x^{(d_x+2)}(x,t)$, $u_t^{(d_t+2)}(x,t) \in C([0,1] \times [0,1])$ with $d_x \geq 1$ and $d_t \geq 1$. Then, the following holds:*

$$\begin{aligned} &||{}_0^C D_t^\beta u(x,t) - {}_0^C\mathfrak{D}_t^\beta u(x,t)||_\infty \\ &\leq C_3 h^{d_x+1} + C_4(2 + H_x ln\ I)2^{d_x-1} H_x^{d_x} \tau^{d_t-1} + C_5 \frac{(R!)^4}{(2R+1)[(2R)!]^3}, \end{aligned}$$

where C_3, C_4 and C_5 are constants.

Proof. Applying triangle inequality, we deduce that

$$\begin{aligned} &||{}_0^C D_t^\beta u(x,t) - {}_0^C\mathfrak{D}_t^\beta u(x,t)||_\infty \\ &\leq ||{}_0^C D_t^\beta u(x,t) - {}_0^C\mathcal{D}_t^\beta u(x,t)||_\infty + ||{}_0^C\mathcal{D}_t^\beta u(x,t) - {}_0^C\mathfrak{D}_t^\beta u(x,t)||_\infty. \end{aligned} \tag{19}$$

By Theorem 1, subtracting Equation (12) from Equation (11) yields

$$
\begin{aligned}
&||{}_0^C D_t^{\beta} u(x,t) - {}_0^C \mathcal{D}_t^{\beta} u(x,t)||_{\infty} \\
&= \frac{1}{\Gamma(2-\beta)} \Big\| \Big(\frac{\partial u(x,0)}{\partial s} - \frac{\partial \mathfrak{R}_{x,I}\mathfrak{R}_{t,K} u(x,0)}{\partial s} \Big) t^{1-\beta} \\
&+ \int_0^t (t-s)^{1-\beta} \Big(\frac{\partial^2 u(x,s)}{\partial s^2} - \frac{\partial^2 \mathfrak{R}_{x,I}\mathfrak{R}_{t,K} u(x,s)}{\partial s^2} \Big) \mathrm{d}s \Big\|_{\infty} \\
&\le \frac{1}{\Gamma(2-\beta)} \Big(\Big\| \frac{\partial u(x,0)}{\partial s} - \frac{\partial \mathfrak{R}_{x,I}\mathfrak{R}_{t,K} u(x,0)}{\partial s} \Big\|_{\infty} \\
&+ \int_0^t (t-s)^{1-\beta} \Big\| \frac{\partial^2 u(x,s)}{\partial s^2} - \frac{\partial^2 \mathfrak{R}_{x,I}\mathfrak{R}_{t,K} u(x,s)}{\partial s^2} \Big\|_{\infty} \mathrm{d}s \Big)
\end{aligned}
$$

Similarly to Theorem 1, by Lemma 1 and $\mathfrak{R}_{t,K} u(x,0) = u(x,0)$, we can deduce that

$$
\begin{aligned}
&\Big\| \frac{\partial u(x,0)}{\partial s} - \frac{\partial \mathfrak{R}_{x,I}\mathfrak{R}_{t,K} u(x,0)}{\partial s} \Big\|_{\infty} \\
&\le \Big\| \frac{\partial u(x,0)}{\partial s} - \frac{\partial \mathfrak{R}_{x,I} u(x,0)}{\partial s} \Big\|_{\infty} + \Big\| \frac{\partial \mathfrak{R}_{x,I} u(x,0)}{\partial s} - \frac{\partial \mathfrak{R}_{x,I}\mathfrak{R}_{t,K} u(x,0)}{\partial s} \Big\|_{\infty} \\
&= \Big\| \frac{\partial u(x,0)}{\partial s} - \frac{\partial \mathfrak{R}_{x,I} u(x,0)}{\partial s} \Big\|_{\infty} \\
&\le C_1 h^{d_x+1}
\end{aligned} \tag{20}
$$

and

$$
\begin{aligned}
&\Big\| \frac{\partial^2 u(x,s)}{\partial s^2} - \frac{\partial^2 \mathfrak{R}_{x,I}\mathfrak{R}_{t,K} u(x,s)}{\partial s^2} \Big\|_{\infty} \\
&\le \Big\| \frac{\partial^2 u(x,s)}{\partial s^2} - \frac{\partial^2 \mathfrak{R}_{x,I} u(x,s)}{\partial s^2} \Big\|_{\infty} + \Big\| \frac{\partial^2 \mathfrak{R}_{x,I} u(x,s)}{\partial s^2} - \frac{\partial^2 \mathfrak{R}_{x,I}\mathfrak{R}_{t,K} u(x,s)}{\partial s^2} \Big\|_{\infty} \\
&\le \Big\| \frac{\partial^2 u(x,s)}{\partial s^2} - \mathfrak{R}_{x,I} \frac{\partial^2 u(x,s)}{\partial s^2} \Big\|_{\infty} + \Big\| \mathfrak{R}_{x,I} \Big\|_{\infty} \Big\| \frac{\partial^2 u(x,s)}{\partial s^2} - \frac{\partial^2 \mathfrak{R}_{t,K} u(x,s)}{\partial s^2} \Big\|_{\infty} \\
&\le C_1 h^{d_x+1} + C_2 (2 + H_x ln\ I) 2^{d_x-1} H_x^{d_x} \tau^{d_t-1}.
\end{aligned} \tag{21}
$$

Combining Equation (20) and Equation (21), we have

$$
\begin{aligned}
&||{}_0^C D_t^{\beta} u(x,t) - {}_0^C \mathcal{D}_t^{\beta} u(x,t)||_{\infty} \\
&\le \frac{1}{\Gamma(2-\beta)} \Big[C_1 t^{1-\beta} h^{d_x+1} + \frac{t^{2-\beta}}{2-\beta} (C_1 h^{d_x+1} + C_2 (2 + H_x ln\ I) 2^{d_x-1} H_x^{d_x} \tau^{d_t-1}) \Big] \\
&\le C_3 h^{d_x+1} + C_4 (2 + H_x ln\ I) 2^{d_x-1} H_x^{d_x} \tau^{d_t-1}.
\end{aligned}
$$

By Lemma 3, if we subtract Equation (13) from Equation (12), then it holds that

$$
\begin{aligned}
&\|{}_0^C \mathcal{D}_t^{\beta} u(x,t) - {}_0^C \mathfrak{D}_t^{\beta} u(x,t)\|_{\infty} \\
&= \Big\| \frac{1}{\Gamma(2-\beta)} \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(x) \Big(\int_0^t (t-s)^{1-\beta} \vartheta_k''(s) u_{ik} \mathrm{d}s - \sum_{r=1}^{R} W_r (t-s_r)^{1-\beta} \vartheta_k''(s_r) u_{ik} \Big) \Big\|_{\infty} \\
&\le \frac{1}{\Gamma(2-\beta)} \frac{(R!)^4}{(2R+1)[(2R)!]^3} \sum_{i=0}^{I} \sum_{k=0}^{K} \phi_i(x) u_{ik} \big\| [(1-s)^{1-\beta} \vartheta_k''(s)]^{(2R)} \big\|_{\infty} \\
&\le C_5 \frac{(R!)^4}{(2R+1)[(2R)!]^3},
\end{aligned}
$$

where C_5 is a constant and $[(1-s)^{1-\beta}\vartheta_k''(s)]^{(2R)}$ is the $2p$-order derivative of $(1-s)^{1-\beta}\vartheta_k''(s)$ with respect to s. According to the above remark and Equation (19), we have

$$||{}_0^C D_t^\beta u(x,t) - {}_0^C\mathfrak{D}_t^\beta u(x,t)||_\infty$$
$$\leq C_3 h^{d_x+1} + C_4(2+H_x ln\ I)2^{d_x-1}H_x^{d_x}\tau^{d_t+1} + C_5\frac{(R!)^4}{(2R+1)[(2R)!]^3}.$$

This proves the theorem. □

4. Numerical Examples

This section demonstrates the effectiveness of the BRICM in solving multi-term time-fractional diffusion problems through four examples. All numerical results are implemented on a AMD Ryzen 5 5600H Windows 10 system by using MATLAB R2022b. The absolute errors E_a and relative errors E_r in all Tables are defined as

$$E_a = \max\{|u_{B_2}(x_i,t_k) - u(x_i,t_k)|\},\ E_r = \frac{\max\{|u_{B_2}(x_i,t_k) - u(x_i,t_k)|\}}{\max\{|u(x_i,t_k)|\}},$$

and the absolute errors in all Figures are denoted by

$$|u_{B_2}(x_i,t_k) - u(x_i,t_k)|,$$

where $u(x_i,t_k)$ is the exact solution and $u_{B_2}(x_i,t_k)$ is the numerical solution, respectively. The convergence order is defined by $\log(E_1/E_2)/(\log(\frac{I_2\times K_2}{I_1\times K_1}))$, where E_2 is the current error and E_1 is the previous error, I_2 and K_2 are the numbers of current nodes, I_1 and K_1 are the numbers of previous nodes.

Example 1. *Consider the following one-term time-fractional convection–diffusion equation:*

$${}_0^C D_t^{0.1}u(x,t) = P(x,t)\frac{\partial^2 u(x,t)}{\partial x^2} - Q(x,t)\frac{\partial u(x,t)}{\partial x} + f(x,t), (x,y) \in [0,1]\times[0,1],$$

with $u(x,0)=0$, $P(x,t) = \frac{128t^{0.2}}{\Gamma(1.9)}$, $Q(x,t) = \frac{64t^{0.1}\cos(\pi x)}{\Gamma(1.9)}$ *and*

$$\begin{aligned} f(x,t) &= \frac{\sin(\pi x)}{\Gamma(1.9)}\Big({}_1F_2(1;[0.95\ 1.45];-\tfrac{\pi^2 t^2}{4})\pi t^{0.9} + 128\pi^2 t^{0.2}\sin(\pi t)\Big) \\ &\quad + \frac{64\pi t^{0.1}}{\Gamma(1.9)}\cos^2(\pi x)\sin(\pi t), \end{aligned}$$

where ${}_1F_2(1;[0.95\ 1.45];-\frac{\pi^2t^2}{4})$ *represents the generalized hypergeometric function. The exact solution of this example is* $u(x,t) = \sin(\pi t)\sin(\pi x)$.

Table 1 shows the E_a and convergence order with $d_x = d_t = I = K$, in which 1000 Gaussian nodes are used. Table 2 shows the comparison of results for Example 1 at different nodes (the second class of Chebyshev nodes and the equidistant nodes), in which $d_x = d_t = I = K$ and the number of Gaussian nodes is 1000. We perceive from these tables that the present scheme maintains the high-order accuracy, which fits well with the theoretical analysis.

Example 2. *Consider the following two-term time-fractional diffusion equation [36]:*

$$\left({}_0^C D_t^\beta + {}_0^C D_t^{\beta_1}\right)u(x,t) = \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t), (x,y)\in[0,1]\times[0,1],$$

with $u(x,0)=0$ *and*

$$f(x,t) = \frac{2t^{2-\beta}}{\Gamma(3-\beta)}\sin(2\pi x) + \frac{2t^{2-\beta_1}}{\Gamma(3-\beta_1)}\sin(2\pi x) + 4\pi^2t^2\sin(2\pi x).$$

The exact solution of this example is $u(x,t) = t^2 \sin(2\pi x)$.

Table 1. E_a and convergence order for Example 1 with $d_x = d_t = I = K$.

$I \times K$	E_a	*Order*	**CPU Time (s)**
4×4	4.5310×10^{-3}	–	0.1089
6×6	2.0196×10^{-5}	6.6753	0.1445
8×8	1.0301×10^{-7}	9.1740	0.2182
10×10	6.4164×10^{-10}	11.3796	0.2408

Table 2. Comparison of results for Example 1 at different nodes with $d_x = d_t = I = K$.

$I \times K$	E_a		E_r	
	Chebyshev Nodes	**Equidistant Nodes**	**Chebyshev Nodes**	**Equidistant Nodes**
5×5	4.7064×10^{-4}	4.0635×10^{-3}	6.0160×10^{-4}	4.4925×10^{-3}
7×7	3.9313×10^{-6}	1.0369×10^{-4}	4.4536×10^{-6}	1.0910×10^{-4}
9×9	2.1562×10^{-8}	1.6026×10^{-6}	2.3250×10^{-8}	1.6524×10^{-6}
11×11	6.2475×10^{-10}	1.6719×10^{-8}	6.5704×10^{-10}	1.7065×10^{-8}

In the second example, 1100 Gaussian nodes are used for numerical calculations. Table 3 shows the absolute errors for Example 2 and compares the present results with the results obtained by the method in [36]. We perceive from Table 3 that the results obtained by the proposed method are more accurate than the results in [36]. Table 4 shows that the second class of Chebyshev nodes is more suitable for this study than the equidistant nodes. Absolute errors and corresponding convergence orders with different fractional orders are listed in Table 5 for Example 2 with $d_x = d_t = I = K$. For the given β, Table 6 shows the relative errors for Example 2 at various time levels with $d_x = d_t = I = K = 14$ and $\beta_1 = 0.1$.

Table 3. Comparison of absolute errors for Example 2 with $d_x = d_t = 4$ and $K = 10$.

I	$\beta = 0.6, \beta_1 = 0.4$		$\beta = 0.9, \beta_1 = 0.1$	
	Present Method	**Method in [36]**	**Present Method**	**Method in [36]**
25	1.6819×10^{-6}	3.5338×10^{-3}	1.6817×10^{-6}	3.5334×10^{-3}
50	6.0084×10^{-8}	8.8537×10^{-4}	5.9549×10^{-8}	8.8528×10^{-4}
100	8.6425×10^{-10}	2.2146×10^{-4}	1.4588×10^{-9}	2.2144×10^{-4}
200	2.0880×10^{-10}	5.5372×10^{-5}	9.5881×10^{-10}	5.5365×10^{-5}

Table 4. Comparison of absolute error of Example 2 at different nodes with $d_x = d_t = I = K$.

$I \times K$	$\beta = 0.6, \beta_1 = 0.4$		$\beta = 0.9, \beta_1 = 0.1$	
	Chebyshev Nodes	**Equidistant Nodes**	**Chebyshev Nodes**	**Equidistant Nodes**
6×6	4.3316×10^{-3}	4.5049×10^{-2}	4.3319×10^{-3}	4.4997×10^{-2}
8×8	1.2194×10^{-4}	4.2148×10^{-3}	1.2193×10^{-4}	4.2113×10^{-3}
10×10	2.3307×10^{-6}	2.3521×10^{-4}	2.3305×10^{-6}	2.3506×10^{-4}
12×12	3.2037×10^{-8}	8.8581×10^{-6}	3.2093×10^{-8}	8.8541×10^{-6}

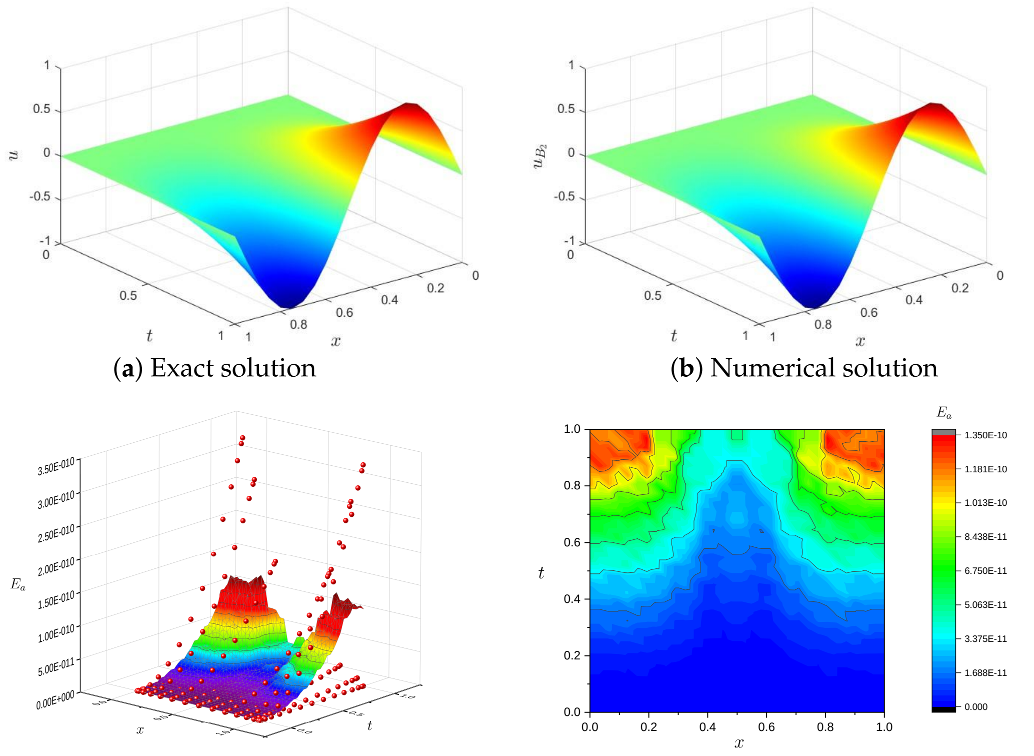
Table 5. E_a and convergence order for Example 2 with $d_x = d_t = I = K$.

$I \times K$	$\beta = 0.2, \beta_1 = 0.1$		$\beta = 0.5, \beta_1 = 0.25$		$\beta = 0.8, \beta_1 = 0.4$	
	E_a	*Order*	E_a	*Order*	E_a	*Order*
6×6	4.3384×10^{-3}	–	4.3340×10^{-3}	–	4.3298×10^{-3}	–
8×8	1.2217×10^{-4}	6.2045	1.2202×10^{-4}	6.2049	1.2187×10^{-4}	6.2053
10×10	2.3337×10^{-6}	8.8686	2.3318×10^{-6}	8.8677	2.3298×10^{-6}	8.8669
12×12	3.2065×10^{-8}	11.7579	3.2047×10^{-8}	11.7572	3.2050×10^{-8}	11.7546

Table 6. E_r for Example 2 at various time levels with $\beta_1 = 0.1$.

	$t = 0.0125$	$t = 0.2831$	$t = 0.5$	$t = 0.7169$	$t = 1$
$\beta = 0.3$	3.4117×10^{-10}	3.4003×10^{-10}	3.3950×10^{-10}	3.3972×10^{-10}	3.3953×10^{-10}
$\beta = 0.4$	3.4392×10^{-10}	3.3966×10^{-10}	3.3948×10^{-10}	3.3950×10^{-10}	3.3955×10^{-10}
$\beta = 0.6$	6.2334×10^{-10}	3.3713×10^{-10}	3.3797×10^{-10}	3.3795×10^{-10}	3.3803×10^{-10}
$\beta = 0.7$	6.8880×10^{-10}	3.2667×10^{-10}	3.3074×10^{-10}	3.3288×10^{-10}	3.3408×10^{-10}

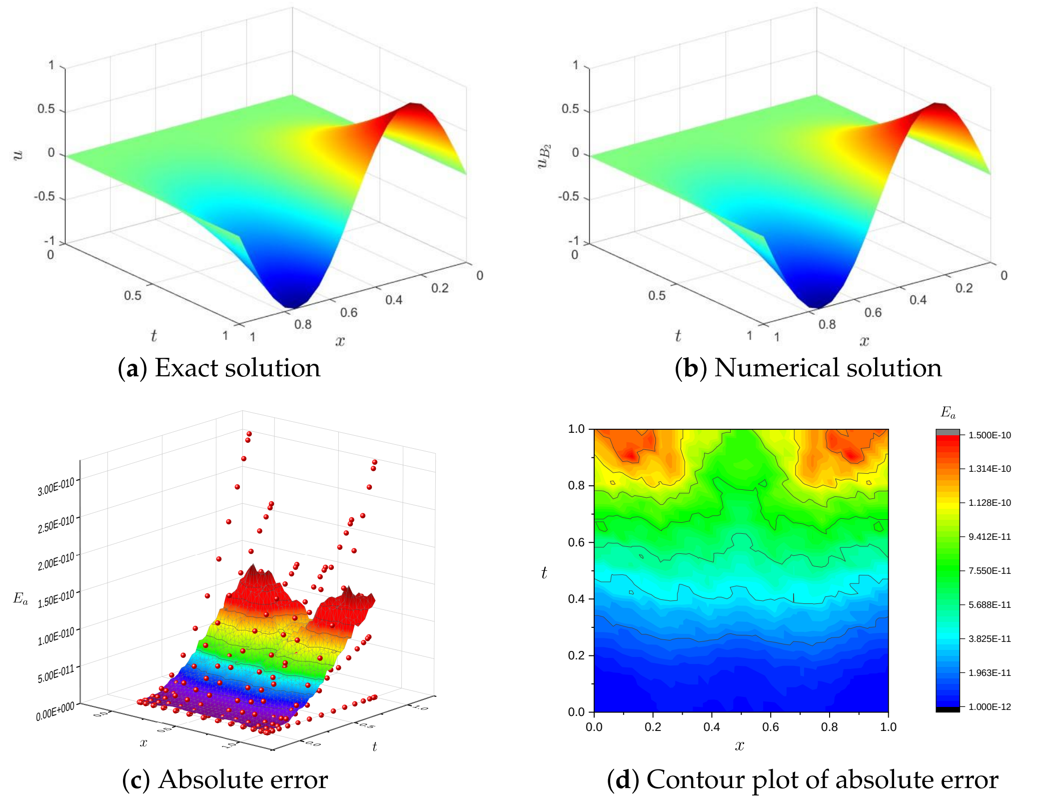
In Figures 1–3, we solve Example 2 by the present method with $d_x = d_t = I = K = 14$ and $p = 1100$. The exact solution, numerical solution, absolute error, and the contour plot of absolute error for $\beta = 0.2,\ \beta_1 = 0.1$ and $\beta = 0.8,\ \beta_1 = 0.4$ are displayed in Figure 1 and Figure 2, respectively. For $\beta = 0.9$ and $\beta_1 = 0.5$, the numerical solution and the situation of solutions at various time levels are shown in Figure 3. We perceive from Figures 1–3 that the numerical solution agrees with the exact solution. Numerical results of the second example show the efficiency and applicability of the present method.



(**a**) Exact solution (**b**) Numerical solution

(**c**) Absolute error (**d**) Contour plot of absolute error

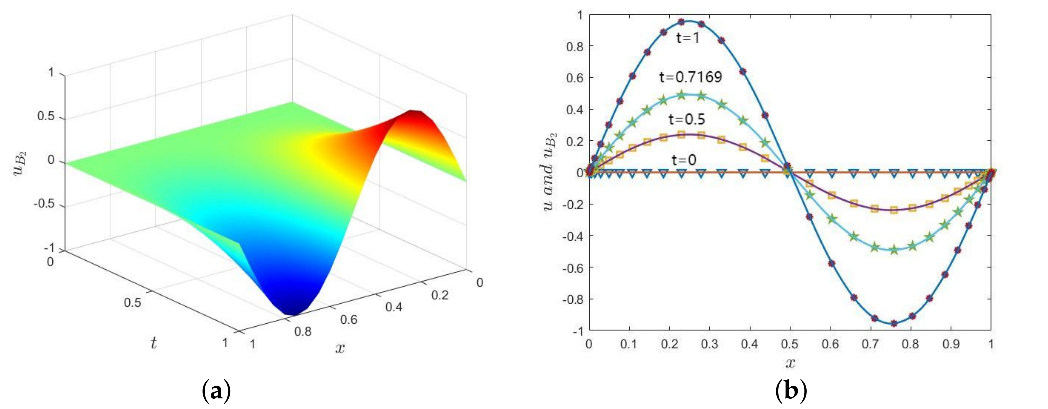
Figure 1. Results of Example 2 with $\beta = 0.2$ and $\beta_1 = 0.1$.



(**a**) Exact solution (**b**) Numerical solution

(**c**) Absolute error (**d**) Contour plot of absolute error

Figure 2. Results of Example 2 with $\beta = 0.8$ and $\beta_1 = 0.4$.



(**a**) (**b**)

Figure 3. Results of Example 2 with $\beta = 0.9$ and $\beta_1 = 0.5$. (**a**) Numerical solution; (**b**) Numerical solution (special symbols) and exact solution (solid line) at various time levels.

Example 3. *Consider the following three-term time-fractional convection–diffusion equation without an exact solution:*

$$\left({}_0^C D_t^{\beta} + \sum_{j=1}^{2} \lambda_j\, {}_0^C D_t^{\beta_j}\right) u(x,t) = P(x,t)\frac{\partial^2 u(x,t)}{\partial x^2} - Q(x,t)\frac{\partial u(x,t)}{\partial x} + f(x,t),$$

with $(x,t) \in [0,1] \times [0,1]$, $u(x,0) = \sin(\pi x)$, $\lambda_1 = \lambda_2 = 1$, $P(x,t) = \frac{16t^{0.2\beta}}{\Gamma(2-\beta)} \sin(\pi x)$, $Q(x,t) = \frac{8t^{0.1\beta}}{\Gamma(2-\beta)} \cos(\pi x)$, *and* $f(x,t)$ *is a given function.*

In the third example, 900 Gaussian nodes and the equidistant nodes are used for numerical calculations. This example is used to test the case where the exact solution is unknown. We will use the solution on the fine grid ($I = K = 16$) as the exact solution. The solutions on coarse grids are used as numerical solutions. For different β, Tables 7 and 8 show the errors for Example 3 with $d_x = d_t = I = K$, $\beta_1 = 0.2$, and $\beta_2 = 0.1$. The calculation results show that the numerical method in our paper has high numerical calculation accuracy.

Table 7. E_a and convergence order for Example 3 with $\beta_1 = 0.2$ and $\beta_2 = 0.1$.

$I \times K$	$\beta = 0.3$		$\beta = 0.5$		$\beta = 0.7$	
	E_a	Order	E_a	Order	E_a	Order
4×4	1.6758×10^{-2}	–	1.6752×10^{-2}	–	1.6778×10^{-2}	–
6×6	4.0572×10^{-4}	4.5885	4.0565×10^{-4}	4.5883	4.0809×10^{-4}	4.5828
8×8	6.4739×10^{-6}	7.1918	6.4784×10^{-6}	7.1903	6.5555×10^{-6}	7.1801
10×10	6.5814×10^{-8}	10.2819	6.6072×10^{-8}	10.2747	6.7627×10^{-8}	10.2491

Table 8. E_r for Example 3 with $\beta_1 = 0.2$ and $\beta_2 = 0.1$ at different β.

$I \times K$	4×4	6×6	8×8	10×10
$\beta = 0.3$	8.3788×10^{-3}	2.0286×10^{-4}	3.2370×10^{-6}	3.2907×10^{-8}
$\beta = 0.5$	8.3758×10^{-3}	2.0282×10^{-4}	3.2392×10^{-6}	3.3036×10^{-8}
$\beta = 0.7$	8.3890×10^{-3}	2.0404×10^{-4}	3.2777×10^{-6}	3.3813×10^{-8}
$\beta = 0.9$	8.4041×10^{-3}	2.0540×10^{-4}	3.3215×10^{-6}	3.2910×10^{-8}

Example 4. *Consider the following four-term time-fractional convection–diffusion equation under the hexagonal region:*

$$\left({}_0^C D_t^{\beta} + \sum_{j=1}^{3} \lambda_j \, {}_0^C D_t^{\beta_j}\right) u(x,t) = \frac{\partial^2 u(x,t)}{\partial x^2} - \frac{\partial u(x,t)}{\partial x} + f(x,t),$$

with $u(x,0) = 0$, $\lambda_1 = 1$, $\lambda_2 = 2$, $\lambda_3 = 0.5$. *The exact solution of this example is* $u(x,t) = (t^3 + t^2)\sin(\pi x)$ *and the corresponding forcing term* $f(x,t)$ *can be obtained by the given conditions.*

In the fourth example, 1330 Gaussian nodes and the second class of Chebyshev nodes are used for numerical calculations. The region where the nodes are located is as follows (see Figure 4):

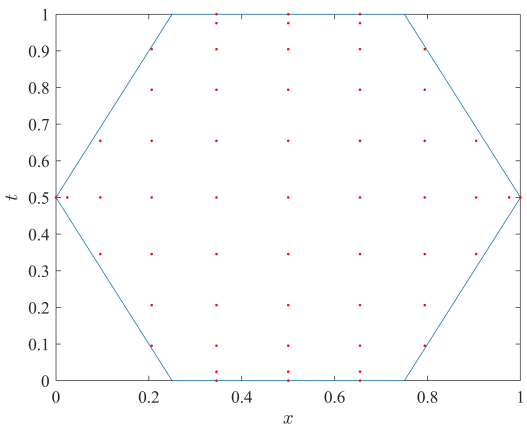


Figure 4. Distribution of solution area and solution nodes.

The results of the fourth example are displayed in Tables 9 and 10 and Figure 5. For different I, K, and β values, Table 9 shows the absolute errors and corresponding convergence orders for Example 4 with $\beta_1 = 0.3$, $\beta_2 = 0.2$, and $\beta_3 = 0.1$. Table 10 shows the relative errors at various time levels. We perceive from these tables that the present method has high computational accuracy and a fast convergence speed. For different β values, the contour plots of absolute error with $I = K = 10$, $\beta_1 = 0.3$, $\beta_2 = 0.2$, and $\beta_3 = 0.1$ are displayed in Figure 5. Numerical results of this example also show the efficiency and applicability of the present method.

Table 9. E_a and convergence order for Example 4 with $\beta_1 = 0.3$, $\beta_2 = 0.2$ and $\beta_3 = 0.1$.

$I \times K$	$\beta = 0.4$		$\beta = 0.6$		$\beta = 0.8$	
	E_a	Order	E_a	Order	E_a	Order
4×4	8.0841×10^{-2}	–	9.0868×10^{-2}	–	1.1011×10^{-1}	–
6×6	2.5106×10^{-4}	4.1655	2.4138×10^{-4}	4.2782	2.1358×10^{-4}	7.7013
8×8	2.2096×10^{-6}	8.2259	2.1925×10^{-6}	8.1711	2.1735×10^{-6}	7.9735
10×10	8.9602×10^{-9}	12.3413	8.6162×10^{-9}	12.4116	1.1141×10^{-8}	11.8163

Table 10. E_r for Example 4 at various time levels with $\beta_1 = 0.3$, $\beta_2 = 0.2$ and $\beta_3 = 0.1$.

	$t = 0.0245$	$t = 0.2061$	$t = 0.5$	$t = 0.7939$	$t = 1$
$\beta = 0.35$	2.1999×10^{-8}	5.2842×10^{-9}	4.2828×10^{-9}	4.4389×10^{-9}	4.4982×10^{-9}
$\beta = 0.55$	3.9612×10^{-8}	6.4776×10^{-9}	4.6389×10^{-9}	4.2809×10^{-9}	4.3786×10^{-9}
$\beta = 0.75$	1.2493×10^{-7}	1.2395×10^{-8}	6.5162×10^{-9}	5.2498×10^{-9}	4.9184×10^{-9}
$\beta = 0.95$	1.5758×10^{-6}	1.0574×10^{-7}	3.1175×10^{-8}	1.7277×10^{-8}	1.3319×10^{-8}

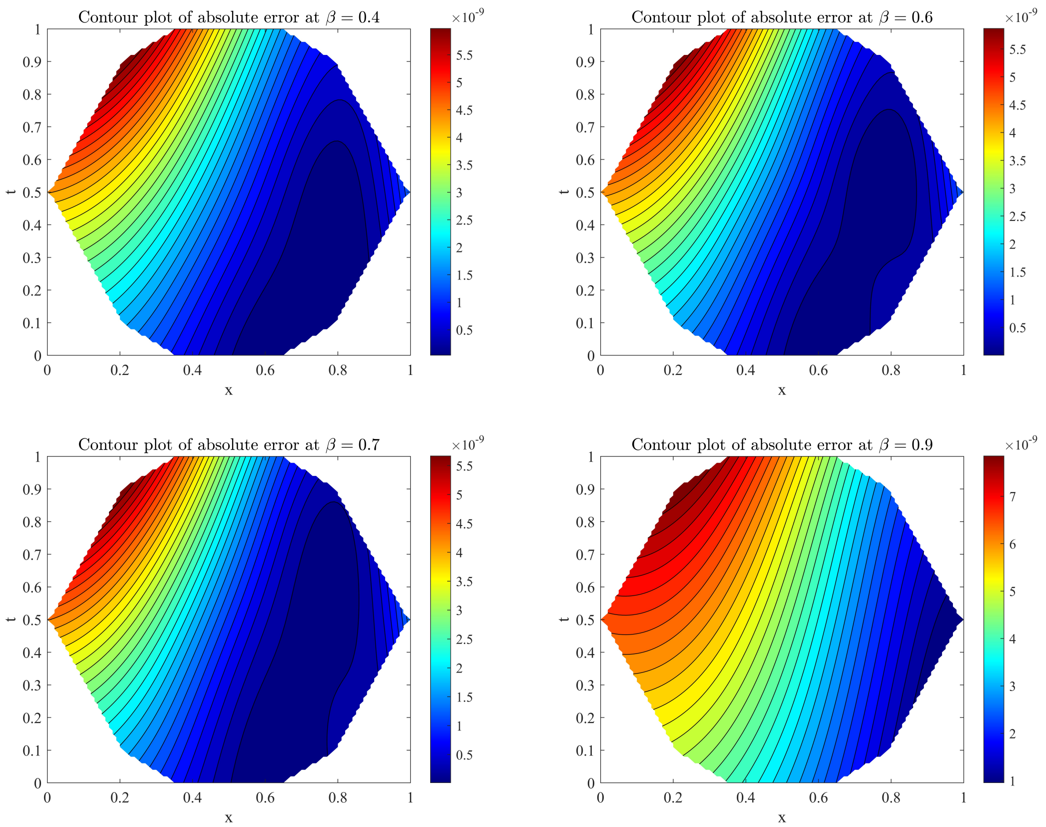


Figure 5. Contour plots of absolute error for Example 4 at different β.

5. Conclusions

In this paper, we give a fully discrete scheme for multi-term time-fractional convection–diffusion equations. The fully discrete scheme is constructed based on the BRICM and the Gauss–Legendre quadrature rule. We prove the convergence of the proposed scheme. The method proposed in this paper can achieve high computational accuracy with very few nodes. We present some numerical examples to illustrate the effectiveness of the method. A comparison of the obtained results with exact solutions and other existing methods reveals that our method is more accurate and efficient for multi-term time-fractional convection–diffusion equations.

Author Contributions: Conceptualization, X.Z.; software, Y.C. and L.W.; formal analysis, X.Z. and Y.C.; writing—original draft preparation, Y.C.; writing—review and editing, X.Z. and S.K.; visualization, Y.C. and L.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Scientific Research Foundation for Talents Introduction of Guizhou University of Finance and Economics (No. 2023YJ16) and the Natural Science Foundation of Xinjiang Uygur Autonomous Region (No. 2022D01E13).

Data Availability Statement: The data analyzed in this study are subject to the following licenses/restrictions: the first author can receive the restrictions. Requests to access these datasets should be directed to cyan19981022@163.com (Y.C., the corresponding author).

Acknowledgments: The authors are very grateful to the referee for carefully reading the article and for many valuable comments.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Fang, Z.W.; Sun, H.W.; Wang, H. A fast method for variable-order Caputo fractional derivative with applications to time-fractional diffusion equations. *Comput. Math. Appl.* **2020**, *80*, 1443–1458. [CrossRef]
2. Mehandiratta, V.; Mehra, M.; Leugering, G. Optimal control problems driven by time-fractional diffusion equations on metric graphs: Optimality system and finite difference approximation. *SIAM J. Control Optim.* **2021**, *59*, 4216–4242. [CrossRef]
3. Gu, X.M.; Sun, H.W.; Zhao, Y.L.; Zheng, X. An implicit difference scheme for time-fractional diffusion equations with a time-invariant type variable order. *Appl. Math. Lett.* **2021**, *120*, 107270. [CrossRef]
4. Zhang, X.D.; Feng, Y.L.; Luo, Z.Y.; Liu, J. A spatial sixth-order numerical scheme for solving fractional partial differential equation. *Appl. Math. Lett.* **2025**, *159*, 109265. [CrossRef]
5. Huang, C.B.; Chen, H.; An, N. β-robust superconvergent analysis of a finite element method for the distributed order time-fractional diffusion equation. *J. Sci. Comput.* **2022**, *90*, 44. [CrossRef]
6. Zheng, X.C.; Wang, H. Optimal-order error estimates of finite element approximations to variable-order time-fractional diffusion equations without regularity assumptions of the true solutions. *IMA J. Numer. Anal.* **2021**, *41*, 1522–1545. [CrossRef]
7. Feng, L.B.; Zhuang, P.; Liu, F.; Turner, I.; Gu, Y. Finite element method for space-time fractional diffusion equation. *Numer. Algorithms* **2016**, *72*, 749–767. [CrossRef]
8. Liu, F.; Zhuang, P.; Turner, I.; Burrage, K.; Anh, V. A new fractional finite volume method for solving the fractional diffusion equation. *Appl. Math. Model.* **2014**, *38*, 3871–3878. [CrossRef]
9. Li, J.; Liu, F.W.; Feng, L.B.; Turner, I. A novel finite volume method for the Riesz space distributed-order diffusion equation. *Comput. Math. Appl.* **2017**, *74*, 772–783. [CrossRef]
10. Lipnikov, K.N.; Svyatskiy, D.; Vassilevski, Y.V. Interpolation-free monotone finite volume method for diffusion equations on polygonal meshes. *J. Comput. Phys.* **2009**, *228*, 703–716. [CrossRef]
11. Zheng, M.L.; Liu, F.W.; Anh, V.V.; Turner, I. A high-order spectral method for the multi-term time-fractional diffusion equations. *Appl. Math. Model.* **2016**, *40*, 4970–4985. [CrossRef]
12. Jafarzadeh, S.; Larios, A.; Bobaru, F. Efficient solutions for nonlocal diffusion problems via boundary-adapted spectral methods. *J. Peridynamics Nonlocal Model.* **2020**, *2*, 85–110. [CrossRef]
13. Kumbinarasaiah, S. Hermite wavelets approach for the multi-term fractional differential equations. *J. Interdiscip. Math.* **2021**, *24*, 1241–1262. [CrossRef]
14. Delkhosh, M.; Parand, K. A new computational method based on fractional Lagrange functions to solve multi-term fractional differential equations. *Numer. Algorithms* **2021**, *88*, 729–766. [CrossRef]
15. Abd-Elhameed, W.M.; Alsuyuti, M.M. Numerical treatment of multi-term fractional differential equations via new kind of generalized Chebyshev polynomials. *Fractal Fract.* **2023**, *7*, 74. [CrossRef]
16. Tural-Polat, S.N.; Dincel, A.T. Numerical solution method for multi-term variable order fractional differential equations by shifted Chebyshev polynomials of the third kind. *Alex. Eng. J.* **2022**, *61*, 5145–5153. [CrossRef]

17. Hashemi, M.S.; Inc, M.; Hajikhah, S. Generalized squared remainder minimization method for solving multi-term fractional differential equations. *Nonlinear Anal. Model.* **2021**, *26*, 57–71. [CrossRef]
18. Shiralashetti, S.C.; Deshi, A.B. An efficient Haar wavelet collocation method for the numerical solution of multi-term fractional differential equations. *Nonlinear Dynam.* **2016**, *83*, 293–303. [CrossRef]
19. Guo, B.L.; Pu, X.K.; Huang, F.H. *Fractional Partial Differential Equations and Their Numerical Solutions*; World Scientific: Singapore, 2015.
20. Hilfer, R. *Applications of Fractional Calculus in Physics*; World Scientific: Singapore, 2000.
21. Podlubny, I. *Fractional Differential Equations*; Academic Press: New York, NY, USA, 1999.
22. Torkaman, S.; Heydari, M.; Loghmani, G.B. A combination of the quasilinearization method and linear barycentric rational interpolation to solve nonlinear multi-dimensional Volterra integral equations. *Math. Comput. Simula.* **2023**, *208*, 366–397.
23. Liu, H.Y.; Ma, Y.Y.; Li, H.; Zhang, W. Combination of discrete technique on graded meshes with barycentric rational interpolation for solving a class of time-dependent partial integro-differential equations with weakly singular kernels. *Comput. Math. Appl.* **2023**, *141*, 159–169.
24. Fahimi-khalilabad, I.; Irandoust-Pakchin, S.; Abdi-Mazraeh, S. High-order finite difference method based on linear barycentric rational interpolation for Caputo type sub-diffusion equation. *Math. Comput. Simula.* **2022**, *199*, 60–80. [CrossRef]
25. Li, J. Linear barycentric rational collocation method for solving biharmonic equation. *Demonstr. Math.* **2022**, *55*, 587–603. [CrossRef]
26. Yang, M.M.; Ma, W.T.; Ge, Y.B. Barycentric rational interpolation method of the Helmholtz equation with irregular domain. *Math. Model. Anal.* **2023**, *28*, 330–351. [CrossRef]
27. Li, J.; Cheng, Y.L. Barycentric rational interpolation method for solving time-dependent fractional convection-diffusion equation. *Electron. Res. Arch.* **2023**, *31*, 4034–4056. [CrossRef]
28. Berrut, J.P.; Mittelmann, H.D. Lebesgue constant minimizing linear rational interpolation of continuous functions over the interval. *Comput. Math. Appl.* **1997**, *33*, 77–86. [CrossRef]
29. Berrut, J.P. Rational functions for guaranteed and experimentallywell-conditioned global interpolation. *Comput. Math. Appl.* **1988**, *15*, 1–16. [CrossRef]
30. Floater, M.S.; Hormann, K.Barycentric rational interpolation with no poles and high rates of approximation. *Numer. Math.* **2007**, *107*, 315–331. [CrossRef]
31. Klein, G.; Berrut, J.P. Linear rational finite differences from derivatives of barycentric rational interpolants. *SIAM J. Numer. Anal.* **2012**, *50*, 643–656. [CrossRef]
32. Kahaner, D.; Moler, C.; Nash, S. *Numerical Methods and Software*; Prentice-Hall, Inc.: Hoboken, NJ, USA, 1989.
33. Fornberg, B. *A practical Guide to Pseudospectral Methods*; Cambridge University Press: Cambridge, UK, 1998.
34. Cirillo, E.; Hormann, K.; Sidon, J. Convergence rates of derivatives of Floater-Hormann interpolants for well-spaced nodes. *Appl. Numer. Math.* **2017**, *116*, 108–118. [CrossRef]
35. Hormann, K.; Klein, G.; Marchi, S.D. Barycentric rational interpolation at quasi-equidistant nodes. *Dolomit. Res. Notes Approx.* **2012**, *5*, 1–6.
36. Ravi Kanth, A.S.V.; Garg, N. An implicit numerical scheme for a class of multi-term time-fractional diffusion equation. *Eur. Phys. J. Plus* **2019**, *134*, 312. [CrossRef]