Networks and Heterogeneous Media

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

NHM, 21(4): 1524–1543.
DOI: 10.3934/nhm.2026058
Received: 04 March 2026
Revised: 28 June 2026
Accepted: 14 July 2026
Published: 17 July 2026

Research article

Analysis of a fully discrete scheme based on the barycentric interpolation method for nonlinear fractional reaction-convection-diffusion equations

Lingna Lu[1,*] **and Leilei Wei**[2]

[1] School of Basic Sciences, Zhengzhou University of Technology, Zhengzhou 450044, China
[2] School of Mathematics and Information Science, North Minzu University, Yinchuan 750021, China

* **Correspondence:** Email: lnl8505@163.com.

Abstract: The primary objective of this study is to develop and analyze a numerical scheme to solve nonlinear fractional reaction-convection-diffusion equations, where the fractional derivative is interpreted in the Caputo sense. First, the nonlinear system is linearized using a direct linearization strategy, after which a barycentric rational interpolation method is employed to construct a fully discrete numerical scheme. A rigorous theoretical framework is established to demonstrate the stability and convergence of the proposed method. To corroborate the theoretical findings, several numerical experiments are conducted, illustrating the method's accuracy, robustness, and computational efficiency. This work advances the numerical analysis of nonlinear systems characterized by fractional-order dynamics and provides effective computational tools to model complex processes in various scientific and engineering contexts.

Keywords: Caputo derivative; barycentric rational interpolation; reaction-convection-diffusion equation

1. Introduction

Reaction–convection–diffusion equations serve as foundational models across various disciplines, including physics [1], chemistry [2], and biology [3]. These models describe how physical quantities evolve in time and space under the combined influence of diffusion (random spreading), convection (directed transport), and reaction (local interactions or transformations). In many classical studies, the diffusion and convection terms are assumed to be linear for analytical simplicity and numerical feasibility [4, 5]. However, real-world systems often exhibit a pronounced nonlinear behavior, particularly in systems with feedback mechanisms, multicomponent interactions, or saturation effects.

These nonlinearities are not merely theoretical curiosities but play a central role in accurately modeling and understanding the complex behaviors of many physical, chemical, and biological

systems. In numerous applications, the underlying processes are inherently nonlinear, and these nonlinearities directly influence both the qualitative and quantitative aspects of the system's evolution [6]. For instance, in fluid dynamics, turbulent flows are characterized by nonlinear interactions across a wide range of spatial and temporal scales, making linear approximations inadequate. Similarly, in chemical kinetics, nonlinear reaction mechanisms such as autocatalysis, inhibition, and saturation often govern the temporal evolution of concentrations, giving rise to phenomena such as oscillatory reactions and pattern formations. In ecological modeling, nonlinear terms naturally appear in predator-prey systems, logistic growth models, and competition dynamics, where population interactions nonlinearly depend on species densities and environmental constraints.

Capturing these complex interactions requires mathematical models that go beyond simple linear approximations. Specifically, nonlinearities in both convective and diffusive terms can significantly affect the stability, propagation speed, and equilibrium states of the solution. Ignoring such effects may lead to models that fail to reproduce key dynamic features, thus resulting in poor predictive power or even qualitative inaccuracies [7].

Fractional calculus has been increasingly employed as a powerful generalization of classical calculus to enhance the descriptive and predictive capabilities of these models. Fractional-order derivatives, as opposed to their integer-order counterparts, are capable of incorporating memory and hereditary properties into the modeling framework [8, 9]. This is particularly advantageous in systems where the future evolution depends not only on the current state but also on the entire past history, which is a common feature in viscoelastic materials, subdiffusive transport, and various biological processes that involve temporal delays or memory feedback.

Among various definitions of fractional derivatives, the Caputo derivative is especially appealing in applied contexts because it allows for the use of standard initial conditions expressed in terms of integer-order derivatives. Mathematically, the Caputo derivative introduces a weighted integral over the past states, where the weights decay in a power-law fashion, thus capturing long-range temporal dependencies [10]. This nonlocality in time provides a natural framework to model anomalous diffusion, where the mean square displacement deviates from the linear time-dependence of classical Brownian motion. Additionally, it proves useful in modeling relaxation processes with memory, such as dielectric relaxation in materials science and adaptation processes in neuronal dynamics.

Moreover, the Caputo formulation retains compatibility with well-established numerical and analytical techniques, which facilitates its integration into existing computational frameworks. As a result, it has been successfully applied in a wide array of disciplines, including physics, geophysics, hydrology, finance, and the control theory. Its ability to interpolate between purely diffusive and wave-like behaviors, depending on the fractional order, provides a flexible tool to model complex systems that exhibit features of both.

In the present work, we consider a nonlinear fractional reaction–convection–diffusion equation (NFRCDE) with a Caputo time-fractional derivative of order $\beta \in (0, 1]$. The model equation is given by the following:

$$ {}_0^C D_t^\beta u = \frac{\partial}{\partial x}\left(P(u)\frac{\partial u}{\partial x}\right) - K_1 u\frac{\partial u}{\partial x} + K_2 u(1-u)Q(u) + f(x,t), \quad (x,t) \in [0,1]\times[0,1], \tag{1.1} $$

subject to the initial and boundary conditions

$$\begin{aligned} &u(x,0) = \psi_1(x), && x \in (0,1),\\ &u(0,t) = \psi_2(t), \quad u(1,t) = \psi_3(t), && t \in (0,1]. \end{aligned}$$

Here, $P(u)$ and $Q(u)$ are nonlinear functions of the unknown $u(x,t)$, and $K_1, K_2 > 0$ quantify the relative strengths of the convective and reactive contributions. The source term $f(x,t)$ and boundary data ψ_1, ψ_2, ψ_3 are assumed to be sufficiently smooth. The Caputo fractional derivative ${}_0^C D_t^\beta u$ is defined as follows:

$${}_0^C D_t^\beta u(x,t) = \begin{cases} \dfrac{1}{\Gamma(1-\beta)} \displaystyle\int_0^t (t-\zeta)^{-\beta} \frac{\partial u(x,\zeta)}{\partial \zeta}\, d\zeta, & 0 < \beta < 1, \\ \dfrac{\partial u(x,t)}{\partial t}, & \beta = 1, \end{cases} \tag{1.2}$$

where $\Gamma(\cdot)$ denotes the Gamma function.

The inclusion of fractional derivatives introduces nonlocality in time. This gives the model the capacity to capture memory effects, such as subdiffusive behavior in porous media and hereditary dynamics in biological tissues. Many researchers have developed numerical approaches for such nonlinear fractional models. For instance, Anley et al. [11] adopted a fully implicit scheme using the Grünwald–Letnikov operator, while Chen et al. [12] proposed an immersed finite element method combined with a nonuniform L1 discretization. Liu [13] studied boundary control techniques for related nonlinear systems using exponential tracking strategies. Further advances in this area can be found in [14–17], where various numerical schemes and theoretical analyses are presented.

Unlike classical interpolation methods that may suffer from oscillations or ill-conditioning, the barycentric rational interpolation (BRI) method introduces precomputed barycentric weights, allowing for stable and efficient interpolation even on nonuniform grids. This has led to successful applications in various settings, including the numerical solution of ordinary and partial differential equations, both linear and nonlinear, as well as fractional models [18–23]. These properties make BRI a natural candidate for the discretization of fractional PDEs with nonlinearities.

In this work, we focus on developing a high-order accurate numerical scheme based on the BRI method to solve the NFRCDE (1.1). The main contributions of this paper are threefold: First, we propose a novel direct linearization strategy that avoids the computational overhead of traditional Newton iterations while retaining high accuracy; second, this strategy is then combined with a Gauss–Legendre quadrature to discretize the fractional integral term; and third, we provide a rigorous theoretical analysis that establishes the stability and convergence of the proposed scheme, and we demonstrate through numerical experiments that our method achieves substantially higher accuracy and lower computational cost compared to existing algorithms.

The rest of this paper is organized as follows: Section 2 introduces the linearization strategy and the fully discrete scheme based on the BRI; Section 3 presents rigorous error estimates for the proposed method; numerical experiments that validate the theoretical convergence are given in Section 4; and finally, conclusions and potential future work are discussed in Section 5.

2. High-accuracy numerical algorithm for Eq (1.1)

2.1. Background and formulation of BRI

In 2007, Floater and Hormann [24] introduced a highly accurate barycentric rational interpolation method that eliminates the occurrence of poles while retaining excellent numerical stability and approximation accuracy. The interpolant takes the following rational barycentric form:

$$r(x) \doteq \frac{\sum_{i=0}^{n} \frac{w_i}{x - x_i} u_i}{\sum_{i=0}^{n} \frac{w_i}{x - x_i}}, \tag{2.1}$$

where $\{x_i\}_{i=0}^{n} \subset [0, 1]$ are equidistant interpolation nodes, $u_i = u(x_i)$, and w_i are the so-called barycentric weights. These weights are defined by the following:

$$w_i = \sum_{z \in O_i} (-1)^z \prod_{\substack{\theta=z \\ \theta \neq i}}^{z+d_x} \frac{1}{x_i - x_\theta}, \quad O_i = \{z \in \mathbb{Z} \cap [0, n] : i - d_x \leq z \leq i\},$$

where $d_x \in \mathbb{N}$ is the degree parameter satisfying $0 \leq d_x \leq n$.

Using Eq (2.1), we define the barycentric rational interpolation function (BRIF) of $u(x)$, denoted by $u_{B_1}(x)$, as follows:

$$u(x) \doteq u_{B_1}(x) = \sum_{i=0}^{n} \Phi_i(x) u_i, \quad \Phi_i(x) = \frac{\frac{w_i}{x - x_i}}{\sum_{g=0}^{n} \frac{w_g}{x - x_g}}. \tag{2.2}$$

This formulation satisfies the interpolation condition

$$\Phi_i(x_\alpha) = \delta_{i\alpha} = \begin{cases} 1, & i = \alpha, \\ 0, & i \neq \alpha, \end{cases} \tag{2.3}$$

thus ensuring exact reconstruction at the nodes.

Analogously, for a set of interpolation points $\{t_j\}_{j=0}^{m} \subset [0, 1]$, we define the BRIF of a function $u(t)$ as follows:

$$u(t) \doteq u_{B_1}(t) = \sum_{j=0}^{m} \Theta_j(t) u_j, \tag{2.4}$$

where $u_j = u(t_j)$, and the basis functions $\Theta_j(t)$ are defined by the following:

$$\Theta_j(t) = \frac{\frac{w_j}{t - t_j}}{\sum_{\rho=0}^{m} \frac{w_\rho}{t - t_\rho}}, \quad w_j = \sum_{\varsigma \in O_j} (-1)^\varsigma \prod_{\substack{v=\varsigma \\ v \neq j}}^{\varsigma+d_t} \frac{1}{t_j - t_v},$$

$$O_j = \{\varsigma \in [0, m] : j - d_t \le \varsigma \le j\}, \quad d_t \in \mathbb{N},\ 0 \le d_t \le m.$$

As before, this interpolation also satisfies $\Theta_j(t_v) = \delta_{jv}$.

To interpolate a bivariate function $u(x, t)$ over a tensor-product grid (x_i, t_j), we denote $u_i(t) := u(x_i, t)$ and $u_{ij} := u(x_i, t_j)$. First, we interpolate in the t-direction using Eq (2.4) to obtain the following:

$$u_i(t) \doteq u_{B_1}(x_i, t) = \sum_{j=0}^{m} \Theta_j(t) u_{ij}. \tag{2.5}$$

Next, applying Eq (2.2) in the x-direction yields the following full two-dimensional BRIF:

$$u(x, t) \doteq \sum_{i=0}^{n} \Phi_i(x) u_i(t) \doteq \sum_{i=0}^{n} \sum_{j=0}^{m} \Phi_i(x) \Theta_j(t) u_{ij} := u_{B_2}(x, t). \tag{2.6}$$

Now, we turn to the derivatives of the interpolant. For any fixed evaluation point (x_a, t_b), the μ-th order partial derivatives of the BRIF $u_{B_2}(x, t)$ are obtained via differentiation of the following basis functions:

$$\frac{\partial^\mu u_{B_2}}{\partial x^\mu}(x_a, t_b) = \sum_{i=0}^{n} \sum_{j=0}^{m} \Phi_i^{(\mu)}(x_a) \Theta_j(t_b) u_{ij} = \sum_{i=0}^{n} \sum_{j=0}^{m} X_{ai}^{(\mu)} \Theta_j(t_b) u_{ij}, \tag{2.7}$$

$$\frac{\partial^\mu u_{B_2}}{\partial t^\mu}(x_a, t_b) = \sum_{i=0}^{n} \sum_{j=0}^{m} \Phi_i(x_a) \Theta_j^{(\mu)}(t_b) u_{ij} = \sum_{i=0}^{n} \sum_{j=0}^{m} \Phi_i(x_a) T_{bj}^{(\mu)} u_{ij}, \tag{2.8}$$

where $X_{ai}^{(\mu)}$ and $T_{bj}^{(\mu)}$ are the entries of the μ-th order barycentric differentiation matrices with respect to x and t, respectively. These matrices are computed recursively as follows:

For the first-order derivative,

$$X_{ai}^{(1)} = \begin{cases} \dfrac{w_i}{w_a} \dfrac{1}{x_a - x_i}, & a \ne i, \\ -\displaystyle\sum_{\substack{l=0 \\ l \ne a}}^{n} X_{al}^{(1)}, & a = i, \end{cases} \qquad T_{bj}^{(1)} = \begin{cases} \dfrac{w_j}{w_b} \dfrac{1}{t_b - t_j}, & b \ne j, \\ -\displaystyle\sum_{\substack{q=0 \\ q \ne b}}^{m} T_{bq}^{(1)}, & b = j. \end{cases}$$

For higher-order derivatives $\mu \ge 2$,

$$X_{ai}^{(\mu)} = \begin{cases} \mu \left(X_{ai}^{(1)} X_{aa}^{(\mu-1)} - \dfrac{X_{ai}^{(\mu-1)}}{x_a - x_i} \right), & a \ne i, \\ -\sum_{\substack{l=0 \\ l \ne a}}^{n} X_{al}^{(\mu)}, & a = i, \end{cases} \qquad T_{bj}^{(\mu)} = \begin{cases} \mu \left(T_{bj}^{(1)} T_{bb}^{(\mu-1)} - \dfrac{T_{bj}^{(\mu-1)}}{t_b - t_j} \right), & b \ne j, \\ -\sum_{\substack{q=0 \\ q \ne b}}^{m} T_{bq}^{(\mu)}, & b = j. \end{cases}$$

For the barycentric rational interpolation with equidistant nodes, the first-order differentiation matrix $\boldsymbol{T}^{(1)}$ satisfies the following error estimate (see [24]): for $u \in C^{d_t+2}[0, T]$:

$$\left| u'(0) - \sum_{j=0}^{m} T_{0j}^{(1)} u(t_j) \right| \le C \tau^{d_t}. \tag{2.9}$$

This ensures that the approximation of the initial derivative term $\left.\frac{\partial u}{\partial t}\right|_{t=0}$ by the first row of $\boldsymbol{T}^{(1)}$ achieves d_t-th order accuracy and is consistent with the overall convergence of the scheme.

2.2. Direct linearized iterative scheme and fully discrete formulation of Eq (1.1)

In this subsection, we construct a fully discrete numerical scheme to solve Eq (1.1) using the BRI method. To this end, we first reformulate the time-fractional derivative using an equivalent integral form that facilitates a numerical approximation.

Under the assumption that $u(x,\cdot) \in C^2[0,T]$ for each fixed x, integrating the Caputo derivative by parts yields the following equivalent representation:

$$\begin{aligned} {}_0^C D_t^\beta u(x,t) &= \frac{1}{\Gamma(1-\beta)} \int_0^t (t-\zeta)^{-\beta} \frac{\partial u(x,\zeta)}{\partial \zeta} d\zeta \\ &= \frac{1}{\Gamma(2-\beta)} \left[(t-\zeta)^{1-\beta} \frac{\partial u}{\partial \zeta} \right]_{\zeta=t}^{\zeta=0} + \frac{1}{\Gamma(2-\beta)} \int_0^t (t-\zeta)^{1-\beta} \frac{\partial^2 u}{\partial \zeta^2} d\zeta \\ &= \frac{t^{1-\beta}}{\Gamma(2-\beta)} \left. \frac{\partial u}{\partial \zeta} \right|_{\zeta=0} + \frac{1}{\Gamma(2-\beta)} \int_0^t (t-\zeta)^{1-\beta} \frac{\partial^2 u}{\partial \zeta^2} d\zeta, \end{aligned}$$

where $\left. \frac{\partial u}{\partial \zeta} \right|_{\zeta=0}$ denotes the partial derivative with respect to time evaluated at $\zeta = 0$. This representation is valid for $0 < \beta < 1$ and requires $u(x,\cdot) \in C^2[0,T]$.

By substituting the above into Eq (1.1), we obtain the following integral form:

$$\begin{aligned} &\frac{t^{1-\beta}}{\Gamma(2-\beta)} \left. \frac{\partial u}{\partial \zeta} \right|_{\zeta=0} + \frac{1}{\Gamma(2-\beta)} \int_0^t (t-\zeta)^{1-\beta} \frac{\partial^2 u}{\partial \zeta^2} \, d\zeta \\ &\qquad = P'(u) \left(\frac{\partial u}{\partial x} \right)^2 + P(u) \frac{\partial^2 u}{\partial x^2} \\ &\qquad\quad - K_1 u \frac{\partial u}{\partial x} + K_2 u(1-u) Q(u) + f(x,t). \end{aligned} \tag{2.10}$$

To handle the nonlinear terms, we apply a direct linearization method. Given an initial approximation u_0, we replace nonlinear terms that involve u with expressions depending on the known function u_0, thus obtaining the following linearized form:

$$\begin{aligned} &\frac{t^{1-\beta}}{\Gamma(2-\beta)} \left. \frac{\partial u}{\partial \zeta} \right|_{\zeta=0} + \frac{1}{\Gamma(2-\beta)} \int_0^t (t-\zeta)^{1-\beta} \frac{\partial^2 u}{\partial \zeta^2} \, d\zeta \\ &\qquad = P'(u_0) \frac{\partial u_0}{\partial x} \frac{\partial u}{\partial x} + P(u_0) \frac{\partial^2 u}{\partial x^2} \\ &\qquad\quad - K_1 u_0 \frac{\partial u}{\partial x} + K_2 u(1-u_0) Q(u_0) + f(x,t). \end{aligned} \tag{2.11}$$

Let $u^{[s]}$ denote the numerical approximation at the s-th iteration. The corresponding direct linearized iterative scheme reads as follows:

$$\begin{aligned} &\frac{t^{1-\beta}}{\Gamma(2-\beta)} \left. \frac{\partial u^{[s]}}{\partial \zeta} \right|_{\zeta=0} + \frac{1}{\Gamma(2-\beta)} \int_0^t (t-\zeta)^{1-\beta} \frac{\partial^2 u^{[s]}}{\partial \zeta^2} \, d\zeta \\ &\qquad = P'(u^{[s-1]}) \frac{\partial u^{[s-1]}}{\partial x} \frac{\partial u^{[s]}}{\partial x} + P(u^{[s-1]}) \frac{\partial^2 u^{[s]}}{\partial x^2} \\ &\qquad\quad - K_1 u^{[s-1]} \frac{\partial u^{[s]}}{\partial x} + K_2 u^{[s]}(1-u^{[s-1]}) Q(u^{[s-1]}) + f(x,t). \end{aligned} \tag{2.12}$$

First, we apply the BRI method in space only, while keeping time continuous. Using the differentiation matrices $\boldsymbol{X}^{(1)}$ and $\boldsymbol{X}^{(2)}$, the spatial derivatives are approximated at the spatial nodes x_i as follows:

$$\frac{\partial u}{\partial x}(x_i,t) \approx \sum_{k=0}^{n} X_{ik}^{(1)} u(x_k,t), \qquad \frac{\partial^2 u}{\partial x^2}(x_i,t) \approx \sum_{k=0}^{n} X_{ik}^{(2)} u(x_k,t).$$

Collecting all spatial nodes x_i ($i = 0,\dots,n$) into the vector $\boldsymbol{u}(t) = [u(x_0,t),\dots,u(x_n,t)]^\top$, the semi-discrete system becomes the following:

$$\begin{aligned} {}_0^C D_t^\beta \boldsymbol{u}(t) &= \operatorname{diag}(P(\boldsymbol{u}(t)))\boldsymbol{X}^{(2)}\boldsymbol{u}(t) + \operatorname{diag}(P'(\boldsymbol{u}(t)) \circ (\boldsymbol{X}^{(1)}\boldsymbol{u}(t)))\boldsymbol{X}^{(1)}\boldsymbol{u}(t) \\ &\quad - K_1 \operatorname{diag}(\boldsymbol{u}(t))\boldsymbol{X}^{(1)}\boldsymbol{u}(t) + K_2\boldsymbol{u}(t) \circ (\mathbf{1} - \boldsymbol{u}(t)) \circ Q(\boldsymbol{u}(t)) + \boldsymbol{f}(t), \end{aligned}$$

where $\circ$ denotes the Hadamard (elementwise) product, $\mathbf{1}$ is the vector of all ones, and $\boldsymbol{f}(t) = [f(x_0,t),\dots,f(x_n,t)]^\top$. This semi-discrete formulation explicitly shows how the spatial operators are discretized while the time-fractional dynamics are preserved in a continuous form.

Introduce the temporal grid $0 = t_0 < t_1 < \cdots < t_m = 1$ with uniform spacing $\tau = 1/m$, and denote $u_{ij} \approx u(x_i,t_j)$. The temporal basis functions $\Theta_j(t)$ and their derivatives are evaluated at these nodes, thus yielding the first-order differentiation matrix $\boldsymbol{T}^{(1)} = [T_{bj}^{(1)}]_{b,j=0}^{m}$ (defined in Section 2.1). In particular, $T_{0j}^{(1)}$ are the entries at the initial time $t_0 = 0$.

The Gauss–Legendre quadrature rule with p nodes $\{\zeta_r\}_{r=1}^{p}$ and weights $\{W_r\}_{r=1}^{p}$ is employed to approximate the fractional integral term. This rule exactly integrates polynomials of degree up to $2p-1$ on $[0,1]$ (see [25] for the convergence analysis of this quadrature for weakly singular integrals).

By substituting the BRI expansions $u(x,t) \approx \sum_{i=0}^{n}\sum_{j=0}^{m} \Phi_i(x)\Theta_j(t)u_{ij}$ into the semi-discrete system and applying the direct linearization strategy (with $u^{[s]}$ denoting the s-th iterate), we obtain the following fully discrete representation:

$$\begin{aligned} &\frac{1}{\Gamma(2-\beta)}\left(t^{1-\beta}\sum_{i=0}^{n}\sum_{j=0}^{m}\Phi_i(x)T_{0j}^{(1)}u_{ij} + \sum_{i=0}^{n}\sum_{j=0}^{m}\sum_{r=1}^{p}(t-\zeta_r)^{1-\beta}\Phi_i(x)\Theta_j''(\zeta_r)W_r u_{ij}\right) \\ &\quad = P'(u_{B_2}^{[s-1]}(x,t)) \cdot \left(\partial_x u_{B_2}^{[s-1]}(x,t)\right) \cdot \left(\partial_x u_{B_2}^{[s]}(x,t)\right) \\ &\qquad + P(u_{B_2}^{[s-1]}(x,t)) \cdot \partial_{xx} u_{B_2}^{[s]}(x,t) - K_1 u_{B_2}^{[s-1]}(x,t) \cdot \partial_x u_{B_2}^{[s]}(x,t) \\ &\qquad + K_2 u_{B_2}^{[s]}(x,t) \cdot \left(1 - u_{B_2}^{[s-1]}(x,t)\right) Q\left(u_{B_2}^{[s-1]}(x,t)\right) + f(x,t), \end{aligned} \tag{2.13}$$

where $T_{0j}^{(1)}$ are the entries of the first-order barycentric differentiation matrix in time evaluated at $t_0 = 0$.

Let us define the discrete temporal integral operator

$$D_j(t_b) := \sum_{r=1}^{p}(t_b - \zeta_r)^{1-\beta}\Theta_j''(\zeta_r)W_r,$$

and the corresponding matrix $\boldsymbol{D} = [D_j(t_b)]_{b,j=0}^{m}$. This matrix represents the discrete approximation of the fractional integral operator.

Then, the fully discrete matrix-vector form for $0 < \beta < 1$ is as follows:

$$\begin{aligned}\Bigg[\frac{1}{\Gamma(2-\beta)}(\boldsymbol{I}_N \otimes \boldsymbol{T}_0^{(1)} + \boldsymbol{I}_N \otimes \boldsymbol{D}) - \operatorname{diag}(P(\boldsymbol{u}^{[s-1]}))(\boldsymbol{X}^{(2)} \otimes \boldsymbol{I}_M) \\ - \operatorname{diag}(P'(\boldsymbol{u}^{[s-1]}) \circ (\boldsymbol{X}^{(1)}\boldsymbol{u}^{[s-1]}))(\boldsymbol{X}^{(1)} \otimes \boldsymbol{I}_M) \\ - K_1 \operatorname{diag}(\boldsymbol{u}^{[s-1]})(\boldsymbol{X}^{(1)} \otimes \boldsymbol{I}_M) \\ - K_2 \operatorname{diag}((1 - \boldsymbol{u}^{[s-1]}) \circ Q(\boldsymbol{u}^{[s-1]}))\Bigg]\boldsymbol{u}^{[s]} = \boldsymbol{F},\end{aligned} \tag{2.14}$$

where the notations are given as follows:

$$\boldsymbol{u}^{[s]} = [\boldsymbol{u}_0^{[s]}, \ldots, \boldsymbol{u}_n^{[s]}]^\top, \quad \boldsymbol{u}_i^{[s]} = [u_{i0}^{[s]}, \ldots, u_{im}^{[s]}]^\top, \quad \boldsymbol{F} = [\boldsymbol{f}_0, \ldots, \boldsymbol{f}_n]^\top,$$

$$\boldsymbol{f}_i = [f(x_i, t_0), \ldots, f(x_i, t_m)]^\top, \quad \boldsymbol{T}_0^{(1)} = \left[t_b^{1-\beta} T_{bj}^{(1)}\right]_{b,j=0}^{m}, \quad \boldsymbol{D} = \left[D_j(t_b)\right]_{b,j=0}^{m},$$

$$\boldsymbol{X}^{(\mu)} = \left[X_{ij}^{(\mu)}\right]_{i,j=0}^{n}, \quad \mu = 1, 2, \quad \boldsymbol{I}_M = \operatorname{diag}(1, \ldots, 1)_{(m+1)\times(m+1)}, \quad \boldsymbol{I}_N = \operatorname{diag}(1, \ldots, 1)_{(n+1)\times(n+1)}.$$

Here, $\boldsymbol{X}^{(1)}\boldsymbol{u}^{[s-1]}$ represents the discrete first derivative $\frac{\partial u^{[s-1]}}{\partial x}$ evaluated at all spatial nodes, and $\operatorname{diag}(P'(\boldsymbol{u}^{[s-1]}) \circ (\boldsymbol{X}^{(1)}\boldsymbol{u}^{[s-1]}))$ is a diagonal matrix whose diagonal entries are $P'(u^{[s-1]}(x_i)) \cdot \frac{\partial u^{[s-1]}}{\partial x}(x_i)$. The source term vector $\boldsymbol{F}$ is assembled from $\boldsymbol{f}(t_b) = [f(x_0, t_b), \ldots, f(x_n, t_b)]^\top$ for each time level t_b, concatenated in the same order as $\boldsymbol{u}^{[s]}$.

For the special case $\beta = 1$, the Caputo derivative reduces to the classical first-order derivative, and the scheme simplifies to the following:

$$\begin{aligned}\Bigg[\boldsymbol{I}_N \otimes \boldsymbol{T}^{(1)} - \operatorname{diag}(P(\boldsymbol{u}^{[s-1]}))(\boldsymbol{X}^{(2)} \otimes \boldsymbol{I}_M) \\ - \operatorname{diag}(P'(\boldsymbol{u}^{[s-1]}) \circ (\boldsymbol{X}^{(1)}\boldsymbol{u}^{[s-1]}))(\boldsymbol{X}^{(1)} \otimes \boldsymbol{I}_M) \\ - K_1 \operatorname{diag}(\boldsymbol{u}^{[s-1]})(\boldsymbol{X}^{(1)} \otimes \boldsymbol{I}_M) \\ - K_2 \operatorname{diag}((1 - \boldsymbol{u}^{[s-1]}) \circ Q(\boldsymbol{u}^{[s-1]}))\Bigg]\boldsymbol{u}^{[s]} = \boldsymbol{F},\end{aligned} \tag{2.15}$$

with $\boldsymbol{T}^{(1)} = \left[T_{bj}^{(1)}\right]_{b,j=0}^{m}$.

The initial and boundary conditions are imposed such that the BRI approximation satisfies the following:

$$u_{B_2}(x, 0) = \psi_1(x), \quad u_{B_2}(0, t) = \psi_2(t), \quad u_{B_2}(1, t) = \psi_3(t).$$

Due to the Kronecker-delta property of barycentric basis functions, this is enforced by setting

$$\begin{aligned} u_{i0}^{[s]} &= \psi_1(x_i), \quad i = 0, 1, \ldots, n, \\ u_{0j}^{[s]} &= \psi_2(t_j), \quad j = 0, 1, \ldots, m, \\ u_{nj}^{[s]} &= \psi_3(t_j), \quad j = 0, 1, \ldots, m. \end{aligned}$$

This completes the construction of the fully discrete iterative BRI-based scheme to solve the NFRCDE (1.1).

3. Error analysis

Throughout the theoretical analysis, we assume that the exact solution $u(x,t)$ possesses sufficient smoothness. More precisely, we assume that

$$u \in C^{d_t+2}([0,T]; C^{d_x+2}(\Omega)), \tag{3.1}$$

where d_x and d_t denote the spatial and temporal interpolation degrees, respectively. This regularity assumption guarantees the existence and boundedness of all derivatives required in the interpolation error estimates and in the analysis of the Caputo fractional derivative.

To analyze the accuracy of the proposed BRI-based scheme, we estimate the interpolation error in both space and time, and further assess the approximation error in the Caputo time-fractional derivative.

Let the spatial step size be $h = \frac{1}{n}$ and the temporal step size be $\tau = \frac{1}{m}$. Denote the grid function values as follows:

$$u_i = u(x_i), \quad u_j = u(t_j), \quad u_{ij} = u(x_i, t_j),$$

which correspond to the spatial and temporal interpolation nodes.

Let $\mathcal{K}_1$ denote the function space spanned by the barycentric basis functions $\{\Phi_i(x)\}_{i=0}^n$ (as defined in Eq (2.2)), and $\mathcal{K}_2$ denote the space spanned by $\{\Theta_j(t)\}_{j=0}^m$ (see Eq (2.4)). Now, we define the corresponding interpolation operators.

Definition 3.1 (Barycentric Interpolation Operators)**.** *Define the one-dimensional interpolation operators*

$$\mathfrak{B}_{x,n} : C[0,1] \to \mathcal{K}_1, \qquad \mathfrak{B}_{t,m} : C[0,1] \to \mathcal{K}_2,$$

such that

$$\mathfrak{B}_{x,n}u(x) = \sum_{i=0}^{n} \Phi_i(x)u_i, \qquad \mathfrak{B}_{t,m}u(t) = \sum_{j=0}^{m} \Theta_j(t)u_j.$$

Define the tensor-product interpolation space $\mathcal{K} = \mathcal{K}_1 \otimes \mathcal{K}_2$ and the two-dimensional interpolation operator as follows:

$$\mathfrak{B}_{x,n}\mathfrak{B}_{t,m}u(x,t) = \sum_{i=0}^{n}\sum_{j=0}^{m} \Phi_i(x)\Theta_j(t)u_{ij},$$

which we denote as the BRI approximation $u_{B_2}(x,t)$.

The interpolation operators are clearly linear, and their approximation quality depends on the smoothness of the target function and the properties of the basis.

Definition 3.2 (Lebesgue Constant)**.** *[26] Let λ_{n,d_x} and λ_{m,d_t} denote the Lebesgue constants associated with the interpolation operators $\mathfrak{B}_{x,n}$ and $\mathfrak{B}_{t,m}$, respectively:*

$$\lambda_{n,d_x} = \max_{x\in[0,1]} \sum_{i=0}^{n} |\Phi_i(x)|, \qquad \lambda_{m,d_t} = \max_{t\in[0,1]} \sum_{j=0}^{m} |\Theta_j(t)|.$$

These constants characterize the stability of the interpolation in the uniform norm.

Next, we introduce key approximation results that quantify the interpolation error and Lebesgue constant growth.

Lemma 3.1 (BRI Approximation Accuracy)**.** *[27] Let the interpolation nodes be equidistant and sufficiently separated. For any integer μ with $0 \le \mu \le \min\{d_x, d_t\}$, under Assumption 3.1, then there exist constants $C_1, C_2 > 0$ such that*

$$|u^{(\mu)}(x) - u_{B_1}^{(\mu)}(x)| \le C_1 h^{d_x+1-\mu}, \qquad |u^{(\mu)}(t) - u_{B_1}^{(\mu)}(t)| \le C_2 \tau^{d_t+1-\mu}.$$

Lemma 3.2 (Growth of Lebesgue Constant)**.** *[28] For equidistant interpolation nodes $\{x_i\}_{i=0}^n$ with a barycentric rational interpolation of degree $d_x \ge 1$, the associated Lebesgue constant satisfies the following logarithmic growth bound:*

$$\lambda_{n,d_x} \le 2^{d_x-1}(2 + \ln n).$$

Now, we present the main interpolation error estimate for the BRI approximation u_{B_2} in two dimensions.

Theorem 3.1. *Under Assumption 3.1, let $u_{B_2}(x,t) = \mathfrak{B}_{x,n}\mathfrak{B}_{t,m}u(x,t)$. Then, there exist constants $C_1, C_2 > 0$ such that*

$$\|u - u_{B_2}\|_\infty \le C_1 h^{d_x+1} + C_2 \lambda_{n,d_x} \tau^{d_t+1},$$

where λ_{n,d_x} is the Lebesgue constant associated with the spatial interpolation operator $\mathfrak{B}_{x,n}$. Moreover, by Lemma 2, $\lambda_{n,d_x} \le 2^{d_x-1}(2 + \ln n)$.

Proof. Let $B_x := B_{x,n}$ and $B_t := B_{t,m}$. Using the identity

$$u - B_x B_t u = (I - B_x)u + B_x(I - B_t)u,$$

we obtain

$$\|u - B_x B_t u\|_\infty \le \|(I - B_x)u\|_\infty + \|B_x(I - B_t)u\|_\infty.$$

By Lemma 3.1, the spatial interpolation error satisfies the following:

$$\|(I - B_x)u\|_\infty \le C_1 h^{d_x+1}.$$

Moreover, by the definition of the Lebesgue constant, the spatial interpolation operator satisfies

$$\|B_x v\|_\infty \le (1 + \lambda_{n,d_x})\|v\|_\infty,$$

for any continuous function v. Hence,

$$\|B_x(I - B_t)u\|_\infty \le (1 + \lambda_{n,d_x})\|(I - B_t)u\|_\infty.$$

Applying Lemma 3.1 again yields

$$\|(I - B_t)u\|_\infty \le C_2 \tau^{d_t+1};$$

therefore,

$$\|B_x(I - B_t)u\|_\infty \le C_2(1 + \lambda_{n,d_x})\tau^{d_t+1}.$$

Combining the above estimates gives the following:

$$\|u - B_x B_t u\|_\infty \le C_1 h^{d_x+1} + C_2(1 + \lambda_{n,d_x})\tau^{d_t+1}.$$

Since

$$1 + \lambda_{n,d_x} \leq C\lambda_{n,d_x},$$

for a generic positive constant C, we finally obtain

$$\|u - u_{B2}\|_\infty \leq C_1 h^{d_x+1} + C_2 \lambda_{n,d_x} \tau^{d_t+1}.$$

This completes the proof.

Remark. Only the spatial Lebesgue constant appears in the final estimate because the temporal interpolation error is directly controlled by Lemma 3.1. The spatial interpolation operator is subsequently applied to this temporal interpolation error, thereby only introducing the spatial stability factor $(1 + \lambda_{n,d_x})$. Therefore, no temporal Lebesgue constant is required in the final error bound.

Next, we examine the numerical error introduced when approximating the Caputo fractional derivative using the BRI interpolation u_{B_2}.

Theorem 3.2. *Let*

$$E_D = {}_0^C D_t^\beta u - {}_0^C D_t^\beta u_{B2},$$

where

$$u_{B2} = B_{x,n} B_{t,m} u.$$

Assume that

$$u \in C^{d_t+1}([0,T]; C^{d_x+1}(\Omega)).$$

Then, there exist constants $C_3, C_4 > 0$ independent of h and τ such that

$$\|E_D\|_\infty \leq C_3 h^{d_x+1} + C_4 \tau^{d_t-1}.$$

Proof. Using the definition of the Caputo fractional derivative,

$${}_0^C D_t^\beta v(t) = \frac{1}{\Gamma(2-\beta)} \left(t^{1-\beta} v_t(0) + \int_0^t (t-\zeta)^{1-\beta} v_{tt}(\zeta)\, d\zeta \right),$$

we obtain

$$\begin{aligned} \|E_D\|_\infty \leq & \frac{1}{\Gamma(2-\beta)} \Bigg(\left\| u_t(0) - u_{B2,t}(0) \right\|_\infty \\ & + \int_0^t (t-\zeta)^{1-\beta} \left\| u_{tt}(\zeta) - u_{B2,tt}(\zeta) \right\|_\infty d\zeta \Bigg). \end{aligned}$$

We estimate the two terms separately.

For the initial derivative, Lemma 3.1 with $\mu = 1$ gives the following:

$$\left\| u_t - u_{B2,t} \right\|_\infty \leq C_1 h^{d_x+1} + C_2 \tau^{d_t}.$$

Next, applying Lemma 3.1 with $\mu = 2$ yields the following:

$$\left\| u_{tt} - u_{B2,tt} \right\|_\infty \leq C_3 h^{d_x+1} + C_4 \tau^{d_t-1}.$$

Therefore,

$$\|E_D\|_\infty \leq \frac{1}{\Gamma(2-\beta)}\Bigg(C_1 h^{d_x+1} + C_2\tau^{d_t} + \int_0^t (t-\zeta)^{1-\beta}\left(C_3 h^{d_x+1} + C_4\tau^{d_t-1}\right)d\zeta\Bigg).$$

Since

$$\int_0^t (t-\zeta)^{1-\beta}d\zeta = \frac{t^{2-\beta}}{2-\beta} \leq \frac{T^{2-\beta}}{2-\beta},$$

we obtain

$$\|E_D\|_\infty \leq \frac{1}{\Gamma(2-\beta)}\Bigg(C_1 h^{d_x+1} + C_2\tau^{d_t} + \frac{T^{2-\beta}}{2-\beta}\left(C_3 h^{d_x+1} + C_4\tau^{d_t-1}\right)\Bigg).$$

Finally, absorbing all coefficients into generic constants yields

$$\|E_D\|_\infty \leq \widetilde{C}_1 h^{d_x+1} + \widetilde{C}_2\tau^{d_t-1},$$

which completes the proof.

4. Numerical examples

In this section, we validate the accuracy and efficiency of the proposed BRI-based numerical scheme by applying it to two benchmark problems that involve NFRCDEs. The examples are designed to test the convergence behavior and error characteristics of the method under various discretization settings.

All numerical experiments were performed using MATLAB R2022b on a standard desktop system equipped with an AMD Ryzen 5 5600H processor and Windows 10 operating system.

To quantify the numerical performance, we define the maximum absolute error and relative error as follows:

$$E_a = \max_{i,j}\left|u(x_i,t_j) - u_{B_2}(x_i,t_j)\right|, \qquad E_r = \frac{\max_{i,j}|u(x_i,t_j) - U_{i,j}|}{\max\{\max_{i,j}|u(x_i,t_j)|, 10^{-12}\}}, \text{respectively,}$$

where $u(x_i,t_j)$ denotes the exact solution, and $u_{B_2}(x_i,t_j)$ represents the BRI-based numerical solution at the spatial-temporal grid point (x_i,t_j).

The visualizations of error distributions in the figures presented below correspond to the following pointwise absolute error:

$$\left|u(x_i,t_j) - u_{B_2}(x_i,t_j)\right|.$$

To assess the convergence behavior, we assume a uniform refinement of the spatial and temporal domains with equal partition sizes (i.e., $m = n$). In this setting, the empirical convergence order is computed using the following standard logarithmic formula:

$$\text{order} = \frac{\log\left(\text{Error}(n_1)/\text{Error}(n_2)\right)}{\log\left(n_2/n_1\right)},$$

where n_1 and n_2 denote successive discretization levels, and Error(n_1), Error(n_2) refer to the corresponding maximum absolute errors.

Unless otherwise specified, all the examples below are computed with the following settings, where the iterative linearization procedure uses a tolerance threshold of 10^{-10} and the initial guess for the nonlinear iteration is taken as the zero function (i.e., $u_0(x,t) \equiv 0$). The specific forms of source terms and exact solutions will be provided within each example to facilitate the verification of accuracy and convergence.

Example 1. Let $\beta = 1$. In this case, the Caputo derivative reduces to the classical first-order time derivative. We consider the following nonlinear reaction–convection–diffusion equation:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} - u\frac{\partial u}{\partial x} + u(1-u)(u-\gamma),$$

posed on the domain $(x,t) \in [0,1] \times [0,1]$, with initial and boundary conditions derived from the exact solution:

$$u(x,t) = \frac{\gamma}{2} + \frac{\gamma}{2}\tanh\left(a_1(x - a_2 t)\right),$$

where the constants are defined as $a_1 = \frac{\gamma}{4}$, $a_2 = \gamma - 1$, and $\gamma = 10^{-3}$.

This example serves as a smooth traveling wave test problem, frequently used to evaluate the performance of numerical methods in the presence of weak convection and nonlinearity. The exact solution exhibits a hyperbolic tangent profile that evolves in time, thus posing a challenge for numerical schemes to accurately resolve steep gradients.

To discretize the problem, we set the number of Gauss–Legendre quadrature nodes as $N_G = 1000$. The spatial interpolation degree and number of subintervals are taken as $d_x = n = 20$, while the temporal interpolation degree and number of subintervals are chosen as $d_t = m = 10$. These values provide a balanced resolution in both space and time.

Table 1 presents the computed maximum absolute error E_a at several selected time levels. For comparison, we also include the corresponding results reported in [30], where DQM stands for the Differential Quadrature Method and SM stands for the Spectral Method. As can be seen, the proposed BRI method achieves substantially higher accuracies under the same discretization settings, which confirms the robustness and efficiency of our scheme.

Table 1. Comparison of E_a with existing algorithms.

(n,m)	t	DQM [30]	SM [30]	Present method
(20, 10)	0.1	1.7366E-06	6.6955E-08	2.8814E-08
	0.2	2.5979E-06	1.0925E-07	4.0119E-08
	0.3	3.4604E-06	1.5155E-07	4.4343E-08
	0.4	4.3241E-06	1.9385E-07	4.5916E-08
	0.5	5.1890E-06	2.3614E-07	4.6503E-08
	0.6	6.0551E-06	2.7843E-07	4.6722E-08
	0.7	6.9224E-06	3.2072E-07	4.6803E-08
	0.8	7.7908E-06	3.6300E-07	4.6833E-08
	0.9	8.6604E-06	4.0527E-07	4.6846E-08
	1	9.5312E-06	4.4755E-07	4.6843E-08
CPU time (seconds)		69.81	57.32	0.2460

Example 2. In the second example, we consider a NFRCDE with a Caputo fractional derivative of order $\beta \in (0, 1)$:

$$ {}_0^C D_t^\beta u = \frac{\partial}{\partial x}\left(u^2 \frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial x}\left(u \frac{\partial u}{\partial x}\right) - 2u\frac{\partial u}{\partial x} + 3u(1-u^2) + \frac{2t^{2-\beta}}{\Gamma(3-\beta)} e^x - 3t^2 e^x, $$

subject to the initial and boundary conditions specified by the exact solution

$$ u(x, t) = t^2 e^x, \quad (x, t) \in [0, 1] \times [0, 1]. $$

Since the spatial and temporal interpolation degrees are chosen to be identical in this example, we denote the following:

$$ d := d_x = d_t. $$

For simplicity, the notation d is used throughout the remainder of this example. This example is specifically designed to assess the proposed algorithm's capability in simultaneously handling several challenging features: A nonlinear source term, the inherent memory effect introduced by the time-fractional derivative, and an exact solution that exhibits exponential growth in space and algebraic growth in time. The presence of both product-type nonlinearities and nonlinear convective derivatives further increases the complexity of the problem, making it a suitable benchmark to evaluate the robustness and accuracy of the numerical scheme.

To approximate the fractional integral term arising from the Caputo derivative, we employ the Gauss–Legendre quadrature rule with $N_G = 1100$ quadrature points, which ensures a sufficiently high resolution for the weakly singular kernel. For simplicity and to facilitate a systematic convergence study, we adopt uniform partitions in both space and time. The interpolation degrees in space and time are taken to be equal (i.e., $d = d_x = d_t$), and the numbers of subintervals are also set equal, with $n = m$. Under this setup, we vary the interpolation degree d to investigate the convergence behavior of the BRI-based method with respect to an increasing approximation order.

Table 2 reports the relative errors E_r and CPU time for different values of β and grid sizes. The results demonstrate that the BRI-based scheme achieves a high accuracy even with coarse discretizations and exhibits efficient runtime performance. For instance, at the coarsest resolution $n = m = 3$, the relative error remains below 10^{-3} for all β, while the CPU time is less than 0.1 seconds, thus indicating an excellent computational efficiency. Tables 3–5 present the maximum absolute error E_a and the corresponding empirical convergence orders for $\beta = 0.1, 0.5,$ and 0.9, respectively. In all cases, the convergence order closely matches $d + 1$, which is consistent with the theoretical prediction in Theorem 3.1. Moreover, as the interpolation degree d increases, the convergence rate improves and the numerical error significantly decreases, thus confirming the high-order accuracy of the proposed method. These numerical results strongly support our theoretical analysis and demonstrate the practical effectiveness of the BRI-based scheme to solve NFRCDEs.

Figure 1 illustrates the comparison between the exact and numerical solutions for three different fractional orders: $\beta = 0.1, 0.5,$ and 0.9. Subplots (a)–(c) show the exact solutions, while subplots (d)–(f) display the corresponding numerical solutions computed via the BRI method. As evident, the BRI solutions exhibit excellent agreement with the exact ones. Figure 2 shows the corresponding absolute error surfaces and their contour plots for the three values of β. The errors are uniformly distributed and remain small across the entire computational domain, thus highlighting the method's

stability and accuracy. These graphical results visually confirm the high fidelity of the BRI scheme in capturing both the qualitative behavior and quantitative accuracy of the exact solution, even for varying fractional orders.

Table 2. E_r and CPU time(s) for Example 2 with $d = n = m$ at different β.

$n \times m$	$\beta = 0.1$		$\beta = 0.5$		$\beta = 0.9$	
	E_r	CPU time(s)	E_r	CPU time(s)	E_r	CPU time(s)
3×3	7.1076E-04	0.0898	7.0890E-04	0.0741	7.0699E-04	0.0690
5×5	3.1719E-06	0.1461	3.1745E-06	0.1074	3.1687E-06	0.1219
7×7	7.6616E-09	0.1697	7.6634E-09	0.2016	7.5650E-09	0.1705
9×9	1.3814E-11	0.2730	1.8372E-11	0.2983	6.8155E-10	0.2579

Table 3. E_a and convergence order for Example 2 with $\beta = 0.1$.

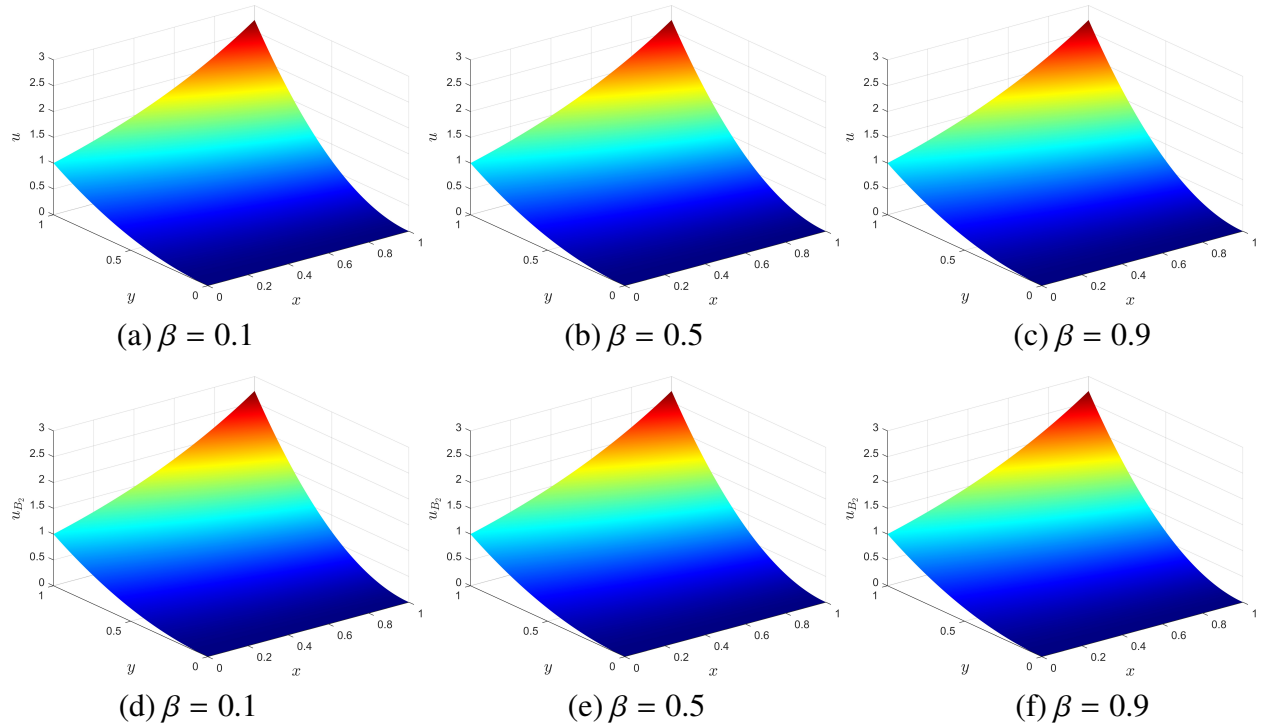
	n	m	E_a	*order*	CPU times(s)
$d = 2$	4	2	3.5400E-03	—	0.0595
	6	3	1.1115E-03	2.8570	0.0748
	8	4	4.7683E-04	2.9418	0.0933
	10	5	2.4669E-04	2.9534	0.1042
$d = 4$	8	4	5.4099E-06	—	0.0883
	10	5	1.8328E-06	4.8506	0.1112
	12	6	7.5176E-07	4.8880	0.1352
	14	7	3.5246E-07	4.9139	0.1625
$d = 6$	12	6	4.2505E-09	—	0.1390
	14	7	1.4796E-09	6.8457	0.1801
	16	8	5.9451E-10	6.8283	0.2383
	18	9	2.6610E-10	6.8250	0.2544

Table 4. E_a and convergence order for Example 2 with $\beta = 0.5$.

	n	m	E_a	*order*	CPU times(s)
$d = 2$	4	2	3.5097E-03	—	0.0624
	6	3	1.1088E-03	2.8418	0.0839
	8	4	4.7723E-04	2.9304	0.0956
	10	5	2.4649E-04	2.9608	0.1137
$d = 3$	6	3	3.8667E-05	—	0.0743
	10	5	6.3564E-06	3.5345	0.1114
	14	7	1.8039E-06	3.7433	0.1572
	18	9	6.8961E-07	3.8262	0.2252
$d = 4$	10	5	1.8314E-06	—	0.1151
	14	7	3.5231E-07	4.8988	0.1608
	18	9	1.0201E-07	4.9318	0.2570
	22	11	3.7782E-08	4.9496	0.4533

Table 5. E_a and convergence order for Example 2 with $\beta = 0.9$.

	n	m	E_a	*order*	CPU times(s)
$d = 2$	6	3	1.1069E-03	—	0.0744
	10	5	2.4622E-04	2.9425	0.1106
	14	7	9.0318E-05	2.9806	0.1776
	18	9	4.2572E-05	2.9928	0.2632
$d = 3$	16	8	1.0834E-06	—	0.2118
	18	9	6.8923E-07	3.8400	0.2691
	20	10	4.5876E-07	3.8634	0.3130
	22	11	3.1721E-07	3.8712	0.4228
$d = 4$	14	7	3.5194E-07	—	0.1637
	18	9	1.0198E-07	4.9288	0.3275
	22	11	3.7710E-08	4.9576	0.3848
	26	13	1.6398E-08	4.9850	1.9755



(a) $\beta = 0.1$ (b) $\beta = 0.5$ (c) $\beta = 0.9$

(d) $\beta = 0.1$ (e) $\beta = 0.5$ (f) $\beta = 0.9$

Figure 1. Comparison between exact and numerical solutions for Example 2 with $\beta = 0.1, 0.5, 0.9$. (a)–(c) Exact solutions, (d)–(f) Numerical solutions.

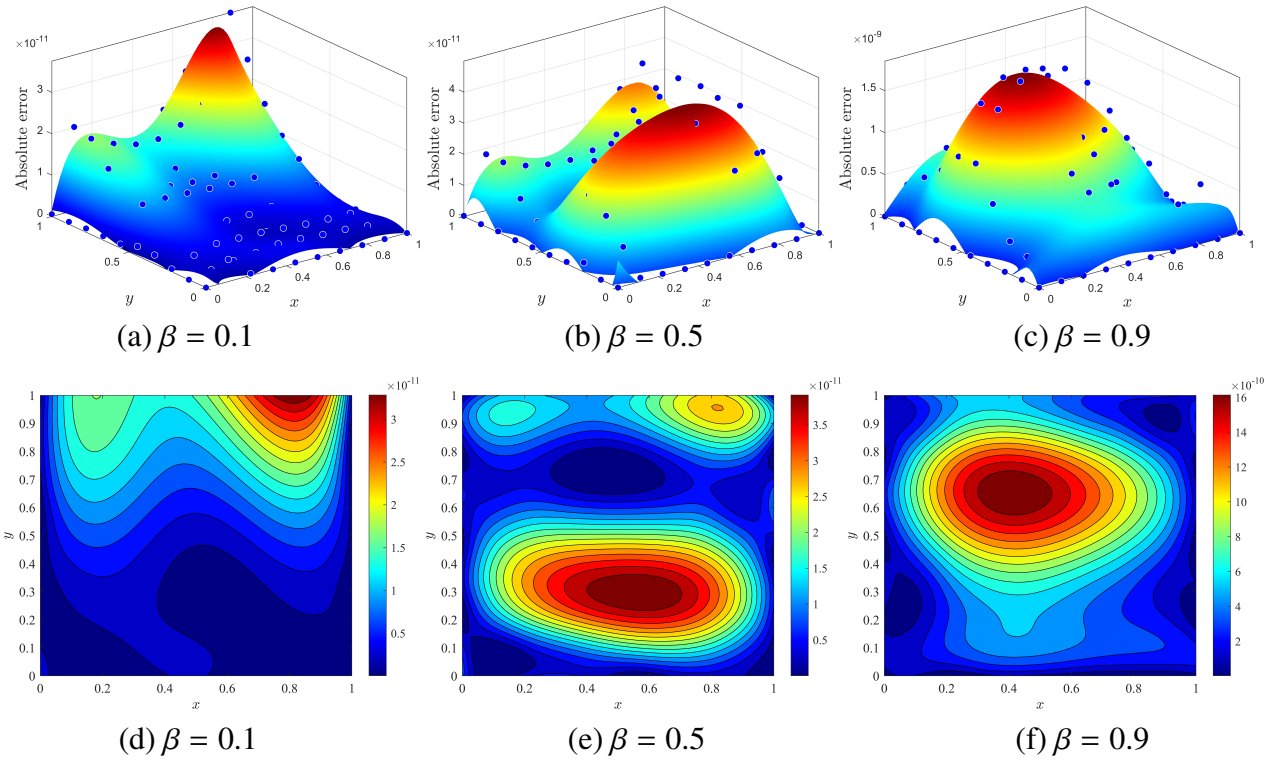


Figure 2. Absolute errors and contour plots for Example 2 with $\beta = 0.1, 0.5, 0.9$. (a)–(c) Absolute errors, (d)–(f) Contour plot of absolute errors.

5. Conclusions

In this paper, we developed and analyzed a fully discrete numerical scheme based on the BRI method for NFRCDEs with a Caputo time-fractional derivative. Rigorous error estimates were established for the proposed scheme. Specifically, Theorem 3.1 showed that the interpolation error satisfies $\|u - u_{B_2}\|_\infty \leq C_1 h^{d_x+1} + C_2(2 + \ln n)2^{d_x-1}\tau^{d_t+1}$, while Theorem 3.2 demonstrated that the error in approximating the Caputo fractional derivative is bounded by $O(h^{d_x+1} + \tau^{d_t-1})$. These results provide a solid theoretical foundation for the BRI-based method, thereby confirming that high-order accuracy can be achieved by increasing the interpolation degrees d_x and d_t.

Benchmark examples were conducted to validate the theoretical findings. For the classical case ($\beta = 1$), Table 1 showed that our method achieved a higher accuracy ($E_a \sim 10^{-8}$) and significantly lower CPU time (0.246 s) compared to the DQM and the SM. For the fractional case ($0 < \beta < 1$), Tables 3–5 confirmed that the empirical convergence order closely matches the theoretical prediction $d + 1$ for $\beta = 0.1, 0.5,$ and 0.9, thus demonstrating robustness across different fractional orders. Figures 1 and 2 further illustrated the excellent agreement between the exact and numerical solutions, with uniformly small errors on the order of 10^{-7} across the entire domain. These numerical results strongly support our theoretical analysis.

In future work, we plan to extend this approach to high-dimensional nonlinear fractional problems and to develop adaptive time-stepping strategies to efficiently handle solutions that exhibit a singular behavior near $t = 0$, such as those with weakly regular initial conditions.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

Leilei Wei is a guest editor for NHM and was not involved in the editorial review or the decision to publish this article. All authors declare that there are no competing interests.

Authors contributions

Lingna Lu: Investigation, Methodology, Programming, Writing–original draft, Project administration. Leilei Wei: Writing–original draft, Review and editing, Visualization, Formal analysis, Revision and editing. All authors read and approved the final manuscript.

References

1. X. X. Dong, W. J. Li, Q. Liu, H. H. Wang, Research on convection-reaction-diffusion model of contaminants in fracturing flowback fluid in non-equidistant fractures with arbitrary inclination of shale gas development, *J. Pet. Sci. Technol.*, **208** (2022), 109479. https://doi.org/10.1016/j.petrol.2021.109479
2. L. Blanco-Cocom, S. Botello-Rionda, L. C. Ordoñez, S. I. Valdez, A reaction-convection-diffusion model for PEM fuel cells, *Finite Elem. Anal. Des.*, **201** (2022), 103703. https://doi.org/10.1016/j.finel.2021.103703
3. V. R. Hosseini, A. A. Mehrizi, H. Karimi-Maleh, M. Naddafi, A numerical solution of fractional reaction-convection-diffusion for modeling PEM fuel cells based on a meshless approach, *Eng. Anal. Boundary Elem.*, **155** (2023), 707–716. https://doi.org/10.1016/j.enganabound.2023.06.016
4. L. Wei, L. Feng, I. Turner, Z. Mao, F. Liu, Numerical investigation of the 2D unsteady natural convection heat transfer equation with tempered fractional constitutive relationship, *Commun. Nonlinear Sci. Numer. Simul.*, **161** (2026), 110071. https://doi.org/10.1016/j.cnsns.2026.110071
5. X. Y. Wang, D. Posny, J. Wang, A reaction-convection-diffusion model for cholera spatial dynamics, *Discrete Contin. Dyn. Syst. - Ser. B*, **21** (2016), 2785–2809. https://doi.org/10.3934/dcdsb.2016073
6. B. L. Guo, X. K. Pu, F. H. Huang, *Fractional Partial Differential Equations and their Numerical Solutions*, World Scientific, 2015. Available from: https://www.gbv.de/dms/tib-ub-hannover/817227784.pdf.
7. I. Podlubny, *Fractional Differential Equations*, Academic Press, New York, 1999.
8. L. Wei, Y. Chen, F. Qin, J. Huang, P. Li, The discontinuous Galerkin method for solving the two-dimensional transport equation with distributed-order memory kernel on triangular meshes, *Appl. Numer. Math.*, **228** (2026), 57–70. https://doi.org/10.1016/j.apnum.2026.05.009

9. Z. H. Gong, C. Y. Liu, B. Wiwatanapataphee, Y. H. Wu, Numerical computation of fractional optimal switched impulsive control problems with time-delay, *J. Comput. Appl. Math.*, **482** (2026), 117315. https://doi.org/10.1016/j.cam.2025.117315
10. X. D. Zhang, Y. L. Feng, L. L. Wei, High-precision numerical simulation for fractional convection-diffusion-reaction equations, *Math. Methods Appl. Sci.*, (2026), 1–18. https://doi.org/10.1002/mma.70756
11. E. F. Anley, M. Basha, A. Hussain, B. X. Dai, Numerical simulation for nonlinear space-fractional reaction convection-diffusion equation with its application, *Alexandria Eng. J.*, **65** (2023), 245–261. https://doi.org/10.1016/j.aej.2022.10.047
12. Y. Chen, Q. Li, H. Yi, Y. Huang, Immersed finite element method for time fractional diffusion problems with discontinuous coefficients, *Comput. Math. Appl.*, **128** (2022), 121–129. https://doi.org/10.1016/j.camwa.2022.09.023
13. W. J. Liu, Exponential tracking and disturbance rejection for nonlinear reaction convection diffusion equations via boundary control, *SIAM J. Control Optim.*, **61** (2023), 151–169. https://doi.org/10.1137/21M1456546
14. M. Basha, E. F. Anley, B. Dai, Numerical solution for a nonlinear time-space fractional convection-diffusion equation, *J. Comput. Nonlinear Dyn.*, **18** (2023), 011006. https://doi.org/10.1115/1.4056218
15. M. H. Heydari, Z. Avazzadeh, A. Atangana, Orthonormal shifted discrete Legendre polynomials for solving a coupled system of nonlinear variable-order time fractional reaction-advection-diffusion equations, *Appl. Numer. Math.*, **161** (2021), 425–436. https://doi.org/10.1016/j.apnum.2020.11.020
16. K. D. Dwivedi, S. Das, Rajeev, D. Baleanu, Numerical solution of highly non-linear fractional order reaction advection diffusion equation using the cubic B-spline collocation method, *Int. J. Nonlinear Sci. Numer. Simul.*, **23** (2022), 1157–1172. https://doi.org/10.1515/ijnsns-2020-0112
17. X. D. Zhang, Y. L. Feng, Z. Y. Luo, J. Liu, A spatial sixth-order numerical scheme for solving fractional partial differential equation, *Appl. Math. Lett.*, **159** (2025), 109265. https://doi.org/10.1016/j.aml.2024.109265
18. S. Torkaman, M. Heydari, G. B. Loghmani, A combination of the quasilinearization method and linear barycentric rational interpolation to solve nonlinear multi-dimensional Volterra integral equations, *Math. Comput. Simul.*, **208** (2023), 366–397. https://doi.org/10.1016/j.matcom.2023.01.039
19. H. Y. Liu, Y. Y. Ma, H. Li, W. Zhang, 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.*, **141** (2023), 159–169. https://doi.org/10.1016/j.camwa.2023.04.018
20. I. F. khalilabad, S. I. Pakchin, S. A. Mazraeh, High-order finite difference method based on linear barycentric rational interpolation for Caputo type sub-diffusion equation, *Math. Comput. Simul.*, **199** (2022), 60–80. https://doi.org/10.1016/j.matcom.2022.03.008
21. J. Li, Linear barycentric rational collocation method for solving biharmonic equation, *Demonstratio Math.*, **55** (2022), 587–603. https://doi.org/10.1515/dema-2022-0151

22. M. M. Yang, W. T. Ma, Y. B. Ge, Barycentric rational interpolation method of the Helmholtz equation with irregular domain, *Math. Model. Anal.*, **28** (2023), 330–351. https://doi.org/10.3846/mma.2023.16408

23. J. Li, Y. L. Cheng, Barycentric rational interpolation method for solving time-dependent fractional convection-diffusion equation, *Electron. Res. Arch.*, **31** (2023), 4034–4056. https://doi.org/10.3934/era.2023205

24. M. S. Floater, K. Hormann, Barycentric rational interpolation with no poles and high rates of approximation, *Numer. Math.*, **107** (2007), 315–331. https://doi.org/10.1007/s00211-007-0093-y

25. X. D. Zhang, Y. Chen, L. L. Wei, High-order numerical approximation for 2D time-fractional advection–diffusion equation under caputo derivative, *Fractal Fract.*, **8** (2024), 474. https://doi.org/10.3390/fractalfract8080474

26. B. Fornberg, *A Practical Guide to Pseudospectral Methods*, Cambridge University Press, 1998.

27. E. Cirillo, K. Hormann, J. Sidon, Convergence rates of derivatives of Floater-Hormann interpolants for well-spaced nodes, *Appl. Numer. Math.*, **116** (2017), 108–118. https://doi.org/10.1016/j.apnum.2016.07.008

28. L. Bos, S. D. Marchi, K. Hormann, G. Klein, On the Lebesgue constant of barycentric rational interpolation at equidistant nodes, *Numer. Math.*, **121** (2012), 461–471. https://doi.org/10.1007/s00211-011-0442-8

29. A. S. V. Ravi Kanth, N. Garg, An implicit numerical scheme for a class of multi-term time-fractional diffusion equation, *Eur. Phys. J. Plus*, **134** (2019), 312. https://doi.org/10.1140/epjp/i2019-12696-8

30. J. Izadian, F. Nateghi, M. Jalili, Comparison of spectral and differential quadrature methods for solving the Burger-Huxley equation, *Commun. Numer. Anal.*, **2013** (2013), 1–7. https://doi.org/10.5899/2013/cna-00171

AIMS Press