International Journal of Innovative Computing, Information and Control
Volume **22**, Number **2**, April **2026**

ICIC International ©2026 ISSN 1349-4198
pp. **521–531**

BARYCENTRIC INTERPOLATION COLLOCATION METHOD TO SOLVE TIME-FRACTIONAL FOKKER-PLANCK EQUATION

Wanjun Song[1] and Jin Li[1,2,*]

[1]School of Science
[2]Computational Intelligence Center
Shandong Jianzhu University
No. 1000, Fengming Road, Lingang Development Zone, Jinan 250101, P. R. China
13791015717@163.com; *Corresponding author: lijin@lsec.cc.ac.cn

Received July 2025; revised October 2025

Abstract. *The time-fractional Fokker-Planck equation (TFFPE) has been formulated to describe complex dynamics, including anomalous diffusion, which is marked by non-locality and memory effect. In this paper, barycentric interpolation collocation methods are used to solve the TFFPE. The fractional derivative part is transformed into a Riemann integral and approximated by using the Gaussian quadrature formula. A corresponding differentiation matrix based on this interpolation method is constructed. Numerical results calculated by barycentric interpolation function are contrasted with the exact solution. The results show that both barycentric Lagrange interpolation collocation method with Chebyshev nodes and barycentric rational interpolation collocation method with equidistant nodes achieve high computational accuracy, and with the increase in the number of nodes, the barycentric Lagrange interpolation collocation method with Chebyshev nodes exhibits higher accuracy.*

Keywords: Barycentric interpolation, Numerical calculation, Time-fractional Fokker-Planck equation, Collocation method

1. **Introduction.** As a fundamental equation in statistical physics [1], the Fokker-Planck equation (FPE) is used to describe the temporal development of particle phase-space probability distributions and is widely used in modeling systems influenced by both deterministic forces and stochastic noise, such as chemical reactions, Brownian motion, and market volatility. As an extension of classical integer order calculus, fractional calculus is characterized by its memory effect, receiving heightened attention in recent years. The fractional models can not only overcome the inconsistency between the experimental and theoretical results of the integer calculus model, but also more accurately describe heredity, memory and path dependence [2]. The inclusion of time-fractional derivative confers memory properties on the system, making system's subsequent temporal development non-Markovian and reliant on its complete past process [3]. TFFPE is employed to describe anomalous diffusion process, which differs from normal Brownian motion due to its memory effects and non-locality [4]. Contrary to the traditional FPE, TFFPE embeds fractional time derivatives, enabling the systems' modeling with sub-diffusion or super-diffusion.

Numerical solutions of TFFPE have been studied by some authors. Deng [5] reformulated TFFPE as Caputo-type fractional ordinary differential equation, and achieved numerical solutions by predictor-corrector approach. Fairweather et al. [6] proposed a numerical method combing orthogonal spline collocation for spatial discretization and

DOI: 10.24507/ijicic.22.02.521

L1-approximated for Caputo fractional derivatives. Chen et al. [7] gave a finite difference approximation, which was extended by Jiang [8] for the TFFPE using Grünwald-Letnikov and L1 formulae for time-fractional derivatives, along with various implicit schemes for spatial terms. Spectral approaches have also been explored, such as [9], where Jacobi polynomials and Fourier-type basis functions were used for temporal and spatial discretization, respectively. More recently, Wei et al. [10] applied a physics-informed neural network with numerical discretization of Caputo derivatives, trained via gradient descent, illustrating the potential of machine learning in solving TFFPE.

The equation under consideration is

$$\begin{cases} \partial_v f = {}_0D_v^{1-\alpha}\left\{\partial_u\left[p(u)f(u,v)\right] + K_\alpha \partial_{uu} f(u,v) + g(u,v)\right\}, & (u,v)\in(a,b)\times(0,V] \\ f(u,0) = f_0(u), & u\in(a,b) \\ f(a,v) = f_a(v),\ f(b,v) = f_b(v), & v\in(0,V] \end{cases}, \quad (1)$$

where $f(u,v)$ is the probability density function characterizing the test particle at a specific position u and a given time v. Over interval $[a,b]$, the function $p(u)$ is non-positive and strictly monotonically decreasing, $K_\alpha > 0$ denotes a diffusion constant, and $g(u,v)$ denotes the source term. In [11], the TFFPE (1) was derived, ${}_0D_v^\alpha$ $(0<\alpha<1)$ is the left Riemann-Liouville fractional derivative,

$${}_0D_v^\alpha f(v) = \frac{d^n}{dv^n} \circ {}_0I_v^{n-\alpha} f(v), \quad \forall n-1 \le \alpha < n, \quad (2)$$

where ${}_0I_v^\alpha f(v)$ is the left Riemann-Liouville fractional integral,

$${}_0I_v^\alpha f(v) = \frac{1}{\Gamma(\alpha)} \int_0^v (v-w)^{\alpha-1} f(w) dw. \quad (3)$$

Let ${}_vI_V^\alpha f(v)$ be the right Riemann-Liouville fractional integral,

$${}_vI_V^\alpha f(v) = \frac{1}{\Gamma(\alpha)} \int_v^V (w-v)^{\alpha-1} f(w) dw. \quad (4)$$

In this paper, we study an equivalent equation of (1) as follows:

$${}_0D_v^\alpha f(u,v) = \partial_u[p(u)f(u,v)] + K_\alpha \partial_{uu} f(u,v) + g(u,v), \quad (5)$$

where ${}_0D_v^\alpha g(v) = \frac{1}{\Gamma(1-\alpha)} \int_0^v (v-w)^{-\alpha} g'(w) dw$, is the Caputo derivative. We equip (5) with the same initial value and boundary constraints as (1). For the proof of the equivalence between (1) and (5), one can refer to [9].

The barycentric interpolation collocation method (BICM) is a collocation method that does not require the division of mesh, with good numerical stability, easy to write computational programs, and with high precision. According to the different selection of base functions, the method can be divided into barycentric Lagrange interpolation collocation method (BLICM) and barycentric rational interpolation collocation method (BRICM). BICM has been widely used in recent years due to its numerical stability and outstanding flexibility in managing both special and equidistant nodal configurations. In the fields of engineering and physics, BICM has been successfully applied to a range of problems, including reaction-diffusion model [12], nonlinear vibration system arises [13], fluid and flow problems [14]. BICM has also been applied to solving the initial boundary value problems governed by both linear and nonlinear partial differential equations, such as ZK-MEW equation [15], telegraph equation [16], Volterra integral differential equation [17], semi-infinite domain problems [18], beam force vibration equation [19], and EFK equation [20]. Previous studies have verified the applicability of the BICM in solving integer-order differential equations and integral equations. However, as the construction of basis functions becomes significantly more complex in higher dimensions, research on its application to

high-dimensional fractional-order equations remains to be further advanced. This study aims to extend the BICM to the solution of TFFPE, which can be regarded as a fractional partial differential equation subject to initial and boundary conditions.

In this paper, the barycentric Lagrange interpolation function (BLIF) and barycentric rational interpolation function (BRIF) are used as the basis functions. Equidistant node and Chebyshev node are used to collocate the (5). The fractional derivative part is transformed into a Riemann integral and is approximated by the Gaussian quadrature formula. The equation is discretized by BICM. And we obtain the matrix equations of (5).

The summary of this paper's construction is as follows. In Section 2, we first outline the theory of BLIF and BRIF. Then we implement BLICM and BRICM. And we obtain the matrix equations of (5). In Section 3, we validate the effectiveness and precision of the method we proposed through a numerical case. In Section 4, we give the summary and conclusion of this paper.

2. Barycentric Interpolation and Matrix Equation of TFFPE.

2.1. Barycentric Lagrange interpolation function.

The BLIF of $f(u)$ at notes $\{u_i\}_{i=0}^m$ is

$$f_m(u) = \sum_{i=0}^{m} \phi_i(u)\, c_i, \tag{6}$$

where $c_i = f(u_i)$, $i = 0, 1, \ldots, m$, and $\{\phi_i(u)\}_{i=0}^m$ is the basis function of the barycentric Lagrange interpolation,

$$\phi_i(u) = \frac{\dfrac{\omega_i}{u - u_i}}{\displaystyle\sum_{k=0}^{m} \frac{\omega_k}{u - u_k}}. \tag{7}$$

Define $\phi_i(u_j) = \delta_{ij}$, where δ_{ij} are Kronecker-delta functions. The weight function ω_i is defined as

$$\omega_i = \frac{1}{\displaystyle\prod_{i=0,\, i\neq k}^{m} (u_i - u_k)}, \quad i = 0, 1, \ldots, m. \tag{8}$$

2.2. Barycentric rational interpolation function.

Compared to traditional polynomial interpolation, barycentric rational interpolation has smoother interpolants and greater stability, and avoids the significant oscillations caused by Runge's phenomenon.

The (9) is defined as the BRIF of $f(u)$ at the nodes $\{u_i\}_{i=0}^m$, when $\{\phi_i(u)\}_{i=0}^m$ is

$$\phi_i(u) = \frac{\dfrac{\omega_i}{u - u_i}}{\displaystyle\sum_{k=0}^{m} \frac{\omega_k}{u - u_k}}, \tag{9}$$

with the interpolation weight

$$\omega_i = \sum_{s\in J_i} (-1)^s \prod_{g=s, g\neq i}^{s+d} \frac{1}{u_i - u_g}, \tag{10}$$

where $J_i = \{s \in I : i - d \leq s \leq i,\ I = \{0, 1, 2, \ldots, m\}\}$ is an index set, and $d \in [0, m]$ is the parameter of barycentric rational interpolation.

The first derivative of (6) is given by $f'_m(u) = \sum_{i=0}^{m} c_i \phi'_i(u)$, where

$$\phi'_i(u_j) = \begin{cases} \dfrac{\omega_i/\omega_j}{u_j - u_i}, & i \neq j \\ -\sum\limits_{i=0, i\neq j}^{m} \phi'_i(u_j), & i = j \end{cases}. \tag{11}$$

The recursive formula for higher order derivatives of the $\phi_i(u)$ at the interpolation nodes is

$$\phi_i^{(k)}(u_j) = \begin{cases} k\left(\phi_j^{(k-1)}(u_j)\phi'_i(u_j) - \dfrac{\phi_i^{(k-1)}(u_j)}{u_j - u_i}\right), & i \neq j,\ k \geq 2 \\ -\sum\limits_{i=0, i\neq j}^{m} \phi_i^{(k)}(u_j), & i = j,\ k \geq 2 \end{cases}. \tag{12}$$

Injecting the interpolation nodes, the k-th derivative of (6) is

$$f^{(k)}(u_j) = \sum_{i=0}^{m} c_i \phi_i^{(k)}(u_j), \quad j = 1, 2, \ldots, m. \tag{13}$$

The matrix forms of (13) can be expressed as

$$f^{(k)} = C^{(k)} f, \tag{14}$$

where $f^{(k)} = \left[f_1^{(k)}, f_2^{(k)}, \ldots, f_m^{(k)}\right]$, $f = [f_1, f_2, \ldots, f_m]$. The elements of $C^{(k)}$ are $C_{ji}^{(k)} = \phi_i^{(k)}(u_j)$.

2.3. **Implementation of BICM and matrix equation for TFFPE.** In this part, for solving TFFPE, we give the matrix equation of BLICM and BRICM. The domain is split into $(m+1)\times(n+1)$ of two kinds of interpolation nodes, equidistant nodes or Chebyshev nodes, i.e., (u_i, v_j), $i = 0, 1, \ldots, m;\ j = 0, 1, \ldots, n$.

Following is the barycentric interpolation function,

$$f(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} L_i(u) V_j(v) f_{ij}, \tag{15}$$

where $f_{ij} = f(u_i, v_j)$ and the interpolation basis functions in spatial and temporal variables are

$$L_i(u) = \frac{\dfrac{\omega_i}{u - u_i}}{\sum\limits_{k=0}^{m} \dfrac{\omega_k}{u - u_k}}, \quad i = 0, 1, \ldots, m. \tag{16}$$

$$V_j(v) = \frac{\dfrac{v_j}{v - v_j}}{\sum\limits_{s=0}^{n} \dfrac{v}{v - v_s}}, \quad j = 0, 1, \ldots, n. \tag{17}$$

The weight functions of BLIF are

$$\omega_i = \frac{1}{\prod_{j=0, j\neq i}^{m} (u_i - u_j)}, \tag{18}$$

$$v_j = \frac{1}{\prod_{k=0, k\neq j}^{n} (v_j - v_k)}, \tag{19}$$

and the weight functions of BRIF are

$$\omega_i = \sum_{s\in J_i} (-1)^s \prod_{g=s, g\neq i}^{s+d_1} \frac{1}{u_i - u_g}, \quad J_i = \{s : i - d_1 \leq s \leq i\}, \tag{20}$$

$$v_j = \sum_{k\in J_j} (-1)^k \prod_{i=k, k\neq j}^{k+d_2} \frac{1}{v_j - v_i}, \quad J_j = \{k : j - d_2 \leq k \leq j\}, \tag{21}$$

where $0 \leq d_1 \leq m$, $0 \leq d_2 \leq n$, $s \in \{0, 1, \ldots, m - d_1\}$ and $k \in \{0, 1, \ldots, n - d_2\}$.

Because the fractional term in (5) is defined as a Caputo fractional derivative of order α, this term exhibits singularity, which can be overcome by performing integration by parts as

$$\begin{aligned} {}_0D_v^\alpha f(u,v) &= \frac{1}{\Gamma(1-\alpha)} \int_0^v (v-w)^{-\alpha} \frac{\partial f(u,w)}{\partial w} dw \\ &= \Gamma_\alpha^1 \left[\frac{\partial f(u,0)}{\partial w} v^{1-\alpha} + \int_0^v (v-w)^{1-\alpha} \frac{\partial^2 f(u,w)}{\partial w^2} dw \right] \end{aligned} \tag{22}$$

where $\Gamma_\alpha^1 = \frac{1}{\Gamma(1-\alpha)(1-\alpha)}$.

Substituting (22) into (5), we obtain

$$\begin{aligned} &\Gamma_\alpha^1 \left[\frac{\partial f(u,0)}{\partial w} v^{1-\alpha} + \int_0^v (v-w)^{1-\alpha} \frac{\partial^2 f(u,w)}{\partial w^2} dw \right] \\ &= \partial_u \left[p(u) f(u,v) \right] + k_\alpha \partial_{uu} f(u,v) + g(u,v). \end{aligned} \tag{23}$$

Combining (15) and (23), one can get the following equation,

$$\begin{aligned} &\Gamma_\alpha^1 \sum_{i=0}^{m} \sum_{j=0}^{n} \left[L_i(u) V_j^{(1)}(0) v^{1-\alpha} \right] f_{ij} + \Gamma_\alpha^1 \sum_{i=0}^{m} \sum_{j=0}^{n} \left[L_i(u) \int_0^v (v-w)^{1-\alpha} V_j^{(2)}(w) dw \right] f_{ij} \\ &= \sum_{i=0}^{m} \sum_{j=0}^{n} \left[P^{(1)}(u) L_i(u) V_j(v) \right] f_{ij} + \sum_{i=0}^{m} \sum_{j=0}^{n} \left[P(u) L_i^{(1)}(u) V_j(v) \right] f_{ij} \\ &\quad + \sum_{i=0}^{m} \sum_{j=0}^{n} \left[K_\alpha L_i^{(2)}(u) V_j(v) \right] f_{ij} + g(u,v). \end{aligned} \tag{24}$$

Let $u = u_s$, $v = v_\theta$, and we get

$$\begin{aligned} &\Gamma_\alpha^1 \sum_{i=0}^{m} \sum_{j=0}^{n} \left[L_i(u_s) V_j^{(1)}(0) v_\theta^{1-\alpha} \right] f_{ij} + \Gamma_\alpha^1 \sum_{i=0}^{m} \sum_{j=0}^{n} \left[L_i(u_s) \int_0^{v_\theta} (v_\theta - w)^{1-\alpha} V_j^{(2)}(w) dw \right] f_{ij} \\ &= \sum_{i=0}^{m} \sum_{j=0}^{n} \left[P^{(1)}(u_s) L_i(u_s) V_j(v_\theta) \right] f_{ij} + \sum_{i=0}^{m} \sum_{j=0}^{n} \left[P(u_s) L_i^{(1)}(u_s) V_j(v_\theta) \right] f_{ij} \end{aligned}$$

$$+\sum_{i=0}^{m}\sum_{j=0}^{n}\left[K_\alpha L_i^{(2)}(u_s)V_j(v_\theta)\right]f_{ij}+g(u_s,v_\theta), \tag{25}$$

where $s=0,1,\ldots,m$, $\theta=0,1,\ldots,n$. The integral items of (25) are formulated as follows:

$$Q_{\theta J}=\int_0^{v_\theta} V_j^{(2)}(w)(v_\theta-w)^{1-\alpha}dw,\quad \theta=j=0,1,\ldots,n, \tag{26}$$

and then we get

$$\begin{aligned}&\Gamma_\alpha^1\sum_{i=0}^{m}\sum_{j=0}^{n}\left[L_i(u_s)V_j^{(1)}(0)v_\theta^{1-\alpha}+L_i(u_s)Q_j(v_\theta)\right]f_{ij}\\ &=g(u_s,v_\theta)+\sum_{i=0}^{m}\sum_{j=0}^{n}\Big[P^{(1)}(u_s)L_i(u_s)V_j(v_\theta)+P(u_s)L_i^{(1)}(u_s)V_j(v_\theta)\\ &\quad+K_\alpha L_i^{(2)}(u_s)V_j(v_\theta)\Big]f_{ij}.\end{aligned} \tag{27}$$

The integral part in (27) is calculated via the application of Gauss quadrature formula with the weight of $\rho(\tau)=(v_\tau-\tau)^{1-\alpha}$, and then we get

$$Q_j(v_\theta)=\int_0^{v_\theta} V_j^{(2)}(v_\theta-w)^{1-\alpha}dw=\sum_{i=1}^{g}V_j^{(2)}(\tau_i)(v_\theta-\tau_i)^{1-\alpha}A_i, \tag{28}$$

where A_i, τ_i denote the Gauss weights and the Gauss points respectively, and g denotes the count of Gauss points in quadrature.

The matrix equation of (29) is derived as follows, and the matrix can be concisely expressed in the following form

$$DF=G, \tag{29}$$

where $F=[f_{00},\ldots,f_{0n},f_{10},\ldots,f_{mm}]^T$, $G=[g_{00},\ldots,g_{0n},g_{10},\ldots,g_{mm}]^T$.

$$\begin{aligned}D=&\Gamma_\alpha^1\left[V\left(I_{m+1}\otimes C_1^{(0,1)}\right)+I_{m+1}\otimes Q\right]-P^{(1)}\left(I_{m+1}\otimes I_{n+1}\right)-P\left(C^{(1,0)}\otimes I_{n+1}\right)\\ &-K\left(C^{(2,0)}\otimes I_{n+1}\right),\end{aligned}$$

$$\operatorname{diag}\left(v_j^{1-\alpha}\right)=\begin{bmatrix}v_0^{1-\alpha}&&&\\&v_1^{1-\alpha}&&\\&&\ddots&\\&&&v_n^{1-\alpha}\end{bmatrix},\ V=\begin{bmatrix}\operatorname{diag}v_j^{1-\alpha}&&&\\&\operatorname{diag}v_j^{1-\alpha}&&\\&&\ddots&\\&&&\operatorname{diag}v_j^{1-\alpha}\end{bmatrix}_{z\times z},$$

where $z=(m+1)\times(n+1)$, $P^{(k)}=\operatorname{diag}\left(p^{(k)}\otimes I_n\right)$, $p^{(k)}=\left[p^{(k)}(u_0),p^{(k)}(u_1),\ldots,p^{(k)}(u_m)\right]^T$, $C^{(P_1,0)}$, $C^{(0,P_2)}$ are differential matrices,

$$C^{(r_1,0)}=\begin{bmatrix}L_{u_0}^{(r_1)}(u_0)&L_{u_1}^{(r_1)}(u_0)&\ldots&L_{u_m}^{(r_1)}(u_0)\\L_{u_0}^{(r_1)}(u_1)&L_{u_1}^{(r_1)}(u_1)&\ldots&L_{u_m}^{(r_1)}(u_1)\\\vdots&\vdots&\vdots&\vdots\\L_{u_0}^{(r_1)}(u_m)&L_{u_1}^{(r_1)}(u_m)&\ldots&L_{u_m}^{(r_1)}(u_m)\end{bmatrix},$$

$$C^{(0,r_2)}=\begin{bmatrix}V_{v_0}^{(r_2)}(v_0)&V_{v_1}^{(r_2)}(v_0)&\ldots&V_{v_n}^{(r_2)}(v_0)\\V_{v_0}^{(r_2)}(v_1)&V_{v_1}^{(r_2)}(v_1)&\ldots&V_{v_n}^{(r_2)}(v_1)\\\vdots&\vdots&\vdots&\vdots\\V_{v_0}^{(r_2)}(v_n)&V_{v_1}^{(r_2)}(v_n)&\ldots&V_{v_n}^{(r_2)}(v_n)\end{bmatrix}.$$

3. **Numerical Results.** To illustrate the accuracy and validity of the methodology, we will present some computational results in Section 3. The absolute error, relative error and maximum absolute error are formulated as follows:

$$E_a = \|f^e - f^c\|_2, \quad E_r = \frac{\|f^e - f^c\|_2}{\|f^e\|_2}, \quad E_\infty = \|f^e - f^c\|_\infty = \max|f^e - f^c|.$$

Among them, the vectors f^e, f^c are denoted as the solutions of numerical and analytical results.

Example 3.1. *Considering the equation*

$$\begin{cases} {}_0D_v^\alpha f(u,v) = f_{uu}(u,v) + g(u,v), & (u,v) \in (0,1)\times(0,1] \\ f(u,0) = 0, & u \in (0,1) \\ f(0,v) = f(1,v) = 0, & v \in (0,1] \end{cases},$$

where

$$g(u,v) = \pi^2 v^6 \sin \pi u + \frac{\Gamma(7) v^{6-\alpha}}{\Gamma(7-\alpha)} \sin \pi u.$$

The exact solution is $f(u,v) = v^6 \sin \pi u$. We adopt the additional method to impose the boundary conditions. Let m and n denote the count of nodes.

Figure 1 presents numerical solution obtained by the BRICM with 10 equidistant nodes and the analytical solution. We can find from Figure 1 that the numerical solution derived via BRICM closely matches the analytical solution.

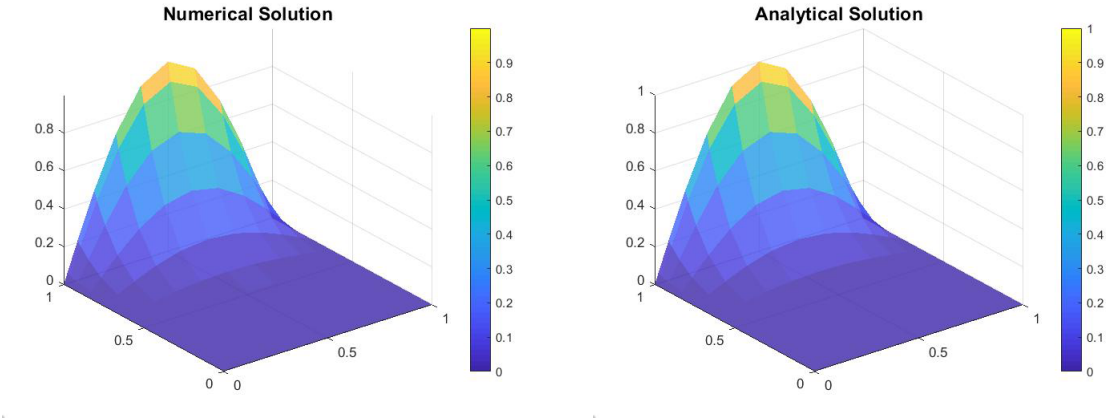
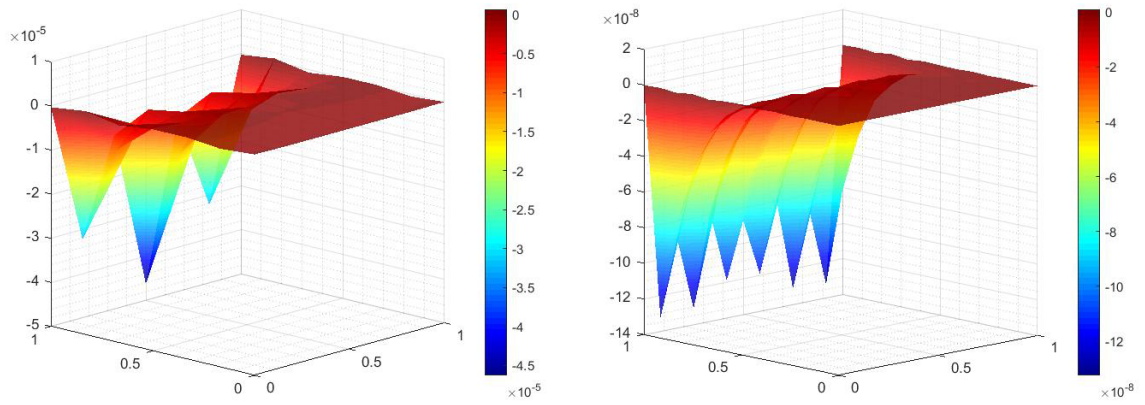


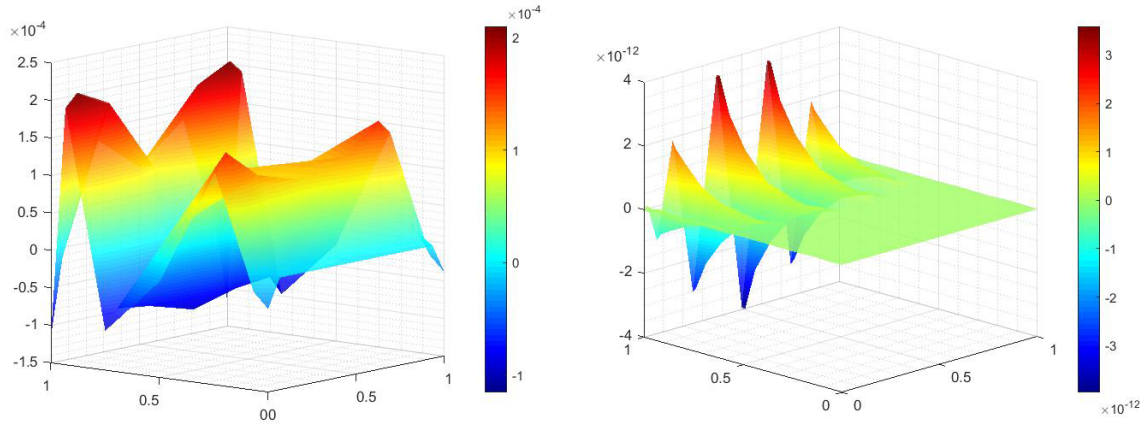
FIGURE 1. Numerical and analytical solutions contrast for Example 3.1

Figures 2 and 3 show the error distributions of BRICM ($\alpha = 0.1$) and BLICM ($\alpha = 0.9$), $m = n = 6, 12$. It can be observed from the figure that as the number of nodes increases, the error accuracy of both methods increases accordingly. When the number of nodes is 12, the error accuracy of BRICM ($\alpha = 0.1$) is 10^{-7}, and that of BLICM ($\alpha = 0.9$) is 10^{-12}. Tables 1, 2, and 3 present the errors of Example 3.1 under different α values when $m = n = 6, 8, 10, 12$ using BRICM with equidistant nodes. Comparing the results, we can find that the errors are gradually decreased beyond the count of nodes, which indicates that the calculation accuracy gradually improves. Tables 4, 5, and 6 present the errors of Example 3.1 under different α values when $m = n = 6, 8, 10, 12$ by using BLICM with Chebyshev nodes. Comparing the results, we can find that the errors are gradually decreased beyond the count of nodes, which indicates that the calculation accuracy gradually improves. Moreover, with the increasing of number of nodes, the method of BLICM with Chebyshev nodes reduces the error more rapidly.



$m = n = 6$ $m = n = 12$

FIGURE 2. $m = n = 6, 12$, $\alpha = 0.1$, errors of BRICM



$m = n = 6$ $m = n = 12$

FIGURE 3. $m = n = 6, 12$, $\alpha = 0.9$, errors of BLICM

TABLE 1. Errors of BRICM with equidistant nodes, $\alpha = 0.1$ for Example 3.1

Number of nodes	E_α	E_r	E_∞
$m = n = 6$	7.0370×10^{-5}	3.8388×10^{-5}	4.6292×10^{-5}
$m = n = 8$	8.7091×10^{-6}	3.9154×10^{-6}	3.4818×10^{-6}
$m = n = 10$	1.6724×10^{-6}	6.3960×10^{-7}	5.6477×10^{-7}
$m = n = 12$	4.4793×10^{-7}	1.4904×10^{-7}	1.3245×10^{-7}

TABLE 2. Errors of BRICM with equidistant nodes, $\alpha = 0.5$ for Example 3.1

Number of nodes	E_α	E_r	E_∞
$m = n = 6$	6.1961×10^{-5}	3.3801×10^{-5}	4.0187×10^{-5}
$m = n = 8$	7.4628×10^{-6}	3.3551×10^{-6}	3.1242×10^{-6}
$m = n = 10$	1.4440×10^{-6}	5.5225×10^{-7}	5.2602×10^{-7}
$m = n = 12$	3.5347×10^{-6}	1.1761×10^{-6}	1.2317×10^{-6}

TABLE 3. Errors of BRICM with equidistant nodes, $\alpha = 0.9$ for Example 3.1

Number of nodes	E_α	E_r	E_∞
$m = n = 6$	1.1811×10^{-4}	6.4431×10^{-5}	3.2072×10^{-5}
$m = n = 8$	1.0492×10^{-5}	4.7169×10^{-6}	2.5713×10^{-6}
$m = n = 10$	2.8260×10^{-6}	1.0808×10^{-6}	6.9409×10^{-7}
$m = n = 12$	1.5746×10^{-5}	5.2392×10^{-6}	6.9339×10^{-6}

TABLE 4. Errors of BLICM with Chebyshev nodes, $\alpha = 0.1$ for Example 3.1

Number of nodes	E_α	E_r	E_∞
$m = n = 6$	9.2176×10^{-5}	5.2674×10^{-5}	4.8310×10^{-5}
$m = n = 8$	6.7882×10^{-7}	3.0419×10^{-7}	3.0878×10^{-7}
$m = n = 10$	3.5319×10^{-9}	1.3031×10^{-9}	1.2222×10^{-9}
$m = n = 12$	4.0024×10^{-11}	1.2556×10^{-11}	1.1035×10^{-11}

TABLE 5. Errors of BLICM with Chebyshev nodes, $\alpha = 0.5$ for Example 3.1

Number of nodes	E_α	E_r	E_∞
$m = n = 6$	1.7478×10^{-4}	9.9878×10^{-4}	6.9791×10^{-5}
$m = n = 8$	6.8391×10^{-7}	3.0647×10^{-7}	3.0878×10^{-7}
$m = n = 10$	3.5390×10^{-9}	1.3057×10^{-9}	1.2290×10^{-9}
$m = n = 12$	1.3658×10^{-11}	4.2845×10^{-12}	4.2710×10^{-12}

TABLE 6. Errors of BLICM with Chebyshev nodes, $\alpha = 0.9$ for Example 3.1

Number of nodes	E_α	E_r	E_∞
$m = n = 6$	7.2280×10^{-4}	4.1305×10^{-4}	2.0991×10^{-4}
$m = n = 8$	1.0415×10^{-6}	4.6672×10^{-7}	3.7843×10^{-7}
$m = n = 10$	3.5733×10^{-9}	1.3183×10^{-9}	3.7843×10^{-7}
$m = n = 12$	1.3385×10^{-11}	4.1989×10^{-12}	3.9695×10^{-12}

Table 7 mainly presents the errors of the two interpolation methods with different interpolation nodes for Example 3.1 when $m = n = 10$ and under different values of α. By comparing the results, it can be seen that, for the same α, the error accuracy of BRICM is 10^{-7}, and that of BLICM is 10^{-9}. Both methods exhibit good error accuracy, but BLICM, as a kind of polynomial interpolation is superior to BRICM in terms of accuracy, and is more suitable for solving the smooth TFFPE.

4. **Conclusion.** In this study, we put forward a novel approach incorporating BLICM and BRICM to get the numerical solution of the TFFPE. The fractional derivative part is transformed into a Riemann integral and approximated by the Gaussian quadrature formula. The method combines barycentric interpolation functions with the Gaussian quadrature rule and discretizes the domain by selecting equidistant nodes and Chebyshev nodes as collocation points within a given interval. By formulating the corresponding differential matrix, the approximate solution is derived. Numerical simulations are used

TABLE 7. E_r of two kinds of interpolation nodes with BRICM and BLICM, when $m = n = 10$, $\alpha = 0.13, 0.20, 0.45, 0.85$ for Example 3.1

α	BRICM		BLICM	
	Equidistant node	Chebyshev node	Equidistant node	Chebyshev node
0.13	5.6335×10^{-7}	4.1642×10^{-7}	3.2693×10^{-9}	1.2235×10^{-9}
0.20	5.5977×10^{-7}	4.1114×10^{-7}	3.2609×10^{-9}	1.2246×10^{-9}
0.45	5.3632×10^{-7}	4.6579×10^{-7}	3.1734×10^{-9}	1.2286×10^{-9}
0.85	4.6260×10^{-7}	1.1680×10^{-6}	2.6740×10^{-9}	1.1964×10^{-9}

to verify the method's reliability and accuracy. Despite these promising results, several limitations of the current work should be noted. The error estimates and convergence analysis for the proposed approach have not been rigorously established. Future research will focus on theoretical analysis of convergence and stability.

Acknowledgment. The work of Jin Li was supported by National Natural Science Foundation of Shandong Province (Grant No. ZR2022MA003), and Doctoral Research Fund of Shandong Jianzhu University (Grant No. X25108).

REFERENCES

[1] R. S. Dubey, B. S. T. Alkahtani and A. Atangana, Analytical solution of space-time fractional Fokker-Planck equation by homotopy perturbation Sumudu transform method, *Mathematical Problems in Engineering*, vol.2015, no.1, 780929, 2015.

[2] X. Lu, *Fractional Reproducing Kernel Collocation Method for Solving Fractional Fokker-Planck Equations*, Ph.D. Thesis, Harbin Normal University, Harbin, China, 2023.

[3] J. Beddrich, E. Süli and B. Wohlmuth, Numerical simulation of the time-fractional Fokker-Planck equation and applications to polymeric fluids, *Journal of Computational Physics*, vol.497, 112598, 2024.

[4] W. Rui, X. Yang and F. Chen, Method of variable separation for investigating exact solutions and dynamical properties of the time-fractional Fokker-Planck equation, *Physica A: Statistical Mechanics and Its Applications*, vol.595, 127068, 2022.

[5] W. Deng, Numerical algorithm for the time fractional Fokker-Planck equation, *Journal of Computational Physics*, vol.227, no.2, pp.1510-1522, 2007.

[6] G. Fairweather, H. Zhang, X. Yang and D. Xu, A backward euler orthogonal spline collocation method for the time-fractional Fokker-Planck equation, *Numerical Methods for Partial Differential Equations*, vol.31, no.5, pp.1534-1550, 2015.

[7] S. Chen, F. Liu, P. Zhuang and V. Anh, Finite difference approximations for the fractional Fokker-Planck equation, *Applied Mathematical Modelling*, vol.33, no.1, pp.256-273, 2009.

[8] Y. Jiang, A new analysis of stability and convergence for finite difference schemes solving the time fractional Fokker-Planck equation, *Applied Mathematical Modelling*, vol.39, no.3, pp.1163-1171, 2015.

[9] Z. Zheng, F. Liu, I. Turner and V. Anh, A novel high order space-time spectral method for the time fractional Fokker-Planck equation, *SIAM Journal on Scientific Computing*, vol.37, no.2, pp.701-724, 2015.

[10] J. L. Wei, G. C. Wu, B. Q. Liu and Z. Zhao, New semi-analytical solutions of the time-fractional Fokker-Planck equation by the neural network method, *Optik*, vol.259, 168896, 2022.

[11] R. Metzler, E. Barkai and J. Klafter, Anomalous diffusion and relaxation close to thermal equilibrium: A fractional Fokker-Planck equation approach, *Physical Review Letters*, vol.82, no.18, 3563, 1999.

[12] Y. Zhang, W. Zhang, C. Zhao and Y. Wang, Numerical solution of a coupled reaction-diffusion model using barycentric interpolation collocation method, *Thermal Science*, vol.24, no.4, pp.2561-2567, 2020.

[13] H. Wu, Y. Wang, W. Zhang and T. Wen, The barycentric interpolation collocation method for a class of nonlinear vibration systems, *Journal of Low Frequency Noise, Vibration and Active Control*, vol.38, nos.3-4, pp.1495-1504, 2019.

[14] Ö. Oruç, An accurate computational method for two-dimensional (2D) fractional Rayleigh-Stokes problem for a heated generalized second grade fluid via linear barycentric interpolation method, *Computers & Mathematics with Applications*, vol.118, pp.120-131, 2022.
[15] Z. Li and J. Li, Numerical solutions for (2 + 1)-dimensional ZK-MEW equation using linear barycentric rational collocation method, *Computers & Mathematics with Applications*, vol.193, pp.332-345, 2025.
[16] J. Li, X. Su and J. Qu, Linear barycentric rational collocation method for solving telegraph equation, *Mathematical Methods in the Applied Sciences*, vol.44, no.14, pp.11720-11737, 2021.
[17] J. Li and Y. Cheng, Linear barycentric rational collocation method for solving second-order Volterra integro-differential equation, *Computational and Applied Mathematics*, vol.39, no.2, 92, 2020.
[18] J. Li, Barycentric rational collocation method for semi-infinite domain problems, *AIMS Mathematics*, vol.8, no.4, pp.8756-8771, 2023.
[19] J. Li and Y. Sang, Linear barycentric rational collocation method for beam force vibration equation, *Shock and Vibration*, vol.2021, no.1, 5584274, 2021.
[20] J. Li and J. Qu, High precision barycentric interpolation collocation method to solve EFK equation, *International Journal of Computer Mathematics*, pp.1-19, 2025.

Author Biography

Wanjun Song received her Bachelor's degree from the School of Science, Shandong Jianzhu University, China, in 2024. She is currently a Master's student in the School of Science, Shandong Jianzhu University, China. Her primary research interests are theory of partial differential equations and barycentric interpolation collocation method.

Jin Li obtained his Ph.D. degree in Academy of Mathematics and System Science from Chinese Academy of Science, China, in 2010; he visited the Shandong University, work as postdoctoral from April 2013 to April 2017.

He is currently a Professor at Shandong Jianzhu University, China. He has been actively engaged in research on the barycentric interpolation collocation method, the numerical approximation of hypersingular integrals, the numerical solution of hypersingular integral equations, the mathematical foundations of machine learning, and quantum computing and so on. He had developed a numerical scheme based on the barycentric interpolation collocation method for solving fractional partial differential equations, accompanied by rigorous error analysis. He also proposed a collocation method utilizing superconvergence points for the efficient solution of hypersingular integral equations. He has authored or co-authored over 100 peer-reviewed academic papers and published a book titled "Approximate Calculation and Application of Hypersingular Integrals" in Science Press.