

A COLLOCATION METHOD FOR COUPLED NONLINEAR SYSTEM OF FRACTIONAL DIFFERENTIAL EQUATIONS

JIN LI^{1,2,*}, WANJUN SONG¹, XIAOHAN YU¹, YANLU LIU¹ AND XU YUAN¹

¹School of Science

²Computational Intelligence Center
Shandong Jianzhu University

No. 1000, Fengming Road, Lingang Development Zone, Jinan 250101, P. R. China
13791015717@163.com; { 2947256273; 1660108173; 1127131159 }@qq.com

*Corresponding author: lijin@lsec.cc.ac.cn

Received December 2025; revised April 2026

ABSTRACT. *Phenomena like interaction of physical fields, population biology, and nonlinear chaotic systems are characterized by coupled nonlinear systems of fractional differential equations (CNFDEs). The numerical solutions of the CNFDEs are discussed by barycentric Lagrange interpolation collocation method (BLICM). First, the fractional derivative terms are transformed into Riemann integrals to eliminate the singularity, followed by approximation using Gaussian quadrature formulas. Next, linearized iterative methods, including the direct linearized iterative method, the partial linearized iterative method and the Newton linearized iterative method, are applied to converting the nonlinear terms into linear terms and linearized iterative schemes are constructed for CNFDEs. Finally, three numerical examples are shown to verify the accuracy of the method. The computational results of three iterative schemes under both equidistant nodes and Chebyshev nodes are compared.*

Keywords: Barycentric Lagrange interpolation, Iterative scheme, Fractional differential equations, Newton linearized iterative scheme

1. Introduction. Fractional differential equations (FDEs), which exhibit memory and hereditary characteristics, are an extension of classical differential equations. FDEs have been widely applied in modeling and analyzing in physical and engineering fields such as electromagnetic wave, power system control, combustion theory, and viscoelastic models [1, 2, 3, 4]. Numerical methods such as the finite difference method [5], variational iteration method [6], finite element method [7], and meshless method [8] are generally used for numerically solving FDEs. The coupled model is a system comprising multiple equations that exhibit interdependent or interactive relationships among one another. Phenomena like interaction of physical fields, population biology, and nonlinear chaotic system are characterized by coupled nonlinear systems of fractional differential equations (CNFDEs). Many researchers have investigated the stability, existence, and uniqueness of solutions of CNFDEs [9, 10, 11]. In [12], a Legendre wavelet-based integral operator matrix is constructed to transform variable-coefficient fractional differential equations into algebraic equations for their solution. In [13], the numerical method based on the operational matrix of Haar wavelets was proposed to solve coupled fractional integro-differential equations. In [14], the Jacobi spectral collocation scheme is utilized to solve CNFDEs.

Within our work, the system with nonlinear terms $w(\cdot)$, $z(\cdot)$ that we focus on is of the form

$$\begin{cases} {}^C_0 D_t^\alpha f(t) = w(f(t)) + u_1(t, g(t)) \\ {}^C_0 D_t^\beta g(t) = z(g(t)) + u_2(t, f(t)) \\ f(0) = f'(0) = 0, g(0) = g'(0) = 0 \end{cases}, \quad (1)$$

where $f(t)$, $g(t)$ are unknown functions, and $u_1(\cdot)$, $u_2(\cdot)$ are known functions. The fractional term in Equation (1) is defined as γ -order Caputo fractional derivative

$${}^C_0 D_t^\gamma f(t) = \frac{1}{\Gamma(1-\gamma)} \int_0^t \frac{\partial f'(\tau)}{\partial \tau} (t-\tau)^{-\gamma} d\tau, \quad 0 < \gamma < 1.$$

The barycentric Lagrange interpolation, developed in [15], addresses the numerical instability and loss of accuracy encountered in the traditional Lagrange interpolation as the number of nodes increases. The barycentric Lagrange interpolation collocation method (BLICM) is a type of meshless numerical method. In recent years, BLICM has been widely applied to the solutions of differential equations due to its strong adaptability to special node distributions and excellent numerical stability. For example, the BLICM was applied to solving fractional pseudo-parabolic equation [16], ZK-MEW equation [17], time fractional advection diffusion equation [18], EFK equation [19] and so on. For the algorithm of solving FDEs using the BLICM, the combination of BLICM with the Gaussian quadrature formula, which has the weight function $\rho(\ell) = (t-\ell)^{\xi-\alpha}$ was proposed in [20] to handle the fractional derivative. Although there have been numerous studies on solving differential equations using the BLICM, studies focusing on the application of this method to the numerical solution of CNFDEs remain relatively scarce.

Based on above, the BLICM is used to solve CNFDEs in this paper, which is an extension of applying BLICM from single fractional differential equations to coupled systems. The advantage of BLICM here is that highly accurate numerical solutions can be obtained with a small number of interpolation nodes. By employing linearized iterative methods, nonlinear problems are transformed into linear equations for solution. Furthermore, the matrix equations and solution efficiencies obtained by the BLICM under three different iterative algorithms, including the direct linearized iterative method, the partial linearized iterative method and the Newton linearized iterative method, are intuitively presented. As the number of nodes increases, the errors calculated by method gradually decrease. The fewer iterations and higher computational efficiency are achieved by Newton linearized iterative method among three iterative methods. Both results are demonstrated through numerical results.

The section arrangement of this paper is as follows. In Section 2, the fractional derivative terms are transformed into Riemann integrals to obtain the equivalent expression of the CNFDEs. In Section 3, the iterative schemes for CNFDEs are constructed using three linearized iterative methods and the corresponding matrix equations are derived based on the BLICM. In Section 4, three numerical examples with different nonlinear terms ($f^2(t)$, $\sin(f(t))$, $f^3(t)$) are given to verify the effectiveness of our method. Section 5 gives summary and conclusion.

2. Transformation of Fractional Part. The α -order Caputo fractional derivative in the first equation of Equation (1) is singular. Integration by parts can be applied to handling the singularity [20], which is

$${}^C_0 D_t^\alpha f(t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{\partial f(\tau)}{\partial \tau} (t-\tau)^{-\alpha} d\tau$$

$$\begin{aligned}
 &= \frac{1}{\Gamma(1-\alpha)(1-\alpha)} \left[\frac{\partial f(0)}{\partial t} t^{1-\alpha} + \int_0^t \frac{\partial^2 f(\tau)}{\partial \tau^2} \frac{d\tau}{(t-\tau)^{\alpha-1}} \right] \\
 &= \mathcal{G}_\alpha \left[\frac{\partial f(0)}{\partial t} t^{1-\alpha} + \int_0^t \frac{\partial^2 f(\tau)}{\partial \tau^2} \frac{d\tau}{(t-\tau)^{\alpha-1}} \right], \tag{2}
 \end{aligned}$$

where $\mathcal{G}_\alpha = \frac{1}{\Gamma(1-\alpha)(1-\alpha)}$. Substituting Equation (2) into the first equation of Equation (1), we obtain

$$\mathcal{G}_\alpha \left[\frac{\partial f(0)}{\partial t} t^{1-\alpha} + \int_0^t \frac{\partial^2 f(\tau)}{\partial \tau^2} \frac{d\tau}{(t-\tau)^{\alpha-1}} \right] = w(f(t)) + u_1(t, g(t)). \tag{3}$$

The same conversion is used for the second equation of Equation (1), in which the fractional term is defined as β -order Caputo fractional derivative. We get

$$\mathcal{G}_\beta \left[\frac{\partial g(0)}{\partial t} t^{1-\beta} + \int_0^t \frac{\partial^2 g(\tau)}{\partial \tau^2} \frac{d\tau}{(t-\tau)^{\beta-1}} \right] = z(g(t)) + u_2(t, f(t)), \tag{4}$$

where $\mathcal{G}_\beta = \frac{1}{\Gamma(1-\beta)(1-\beta)}$.

The equivalent equations of Equation (1) are represented as follows:

$$\begin{cases} D_f^\alpha = w(f(t)) + u_1(t, g(t)) \\ D_g^\beta = z(g(t)) + u_2(t, f(t)) \end{cases}, \tag{5}$$

where $D_f^\alpha = \mathcal{G}_\alpha \left[\frac{\partial f(0)}{\partial t} t^{1-\alpha} + \int_0^t \frac{\partial^2 f(\tau)}{\partial \tau^2} \frac{d\tau}{(t-\tau)^{\alpha-1}} \right]$, $D_g^\beta = \mathcal{G}_\beta \left[\frac{\partial g(0)}{\partial t} t^{1-\beta} + \int_0^t \frac{\partial^2 g(\tau)}{\partial \tau^2} \frac{d\tau}{(t-\tau)^{\beta-1}} \right]$. Subsequently, we focus on the equivalent form of Equation (1), which is represented by Equation (5) and shares the same initial and boundary conditions as Equation (1).

3. Linearized Iterative Scheme and the Matrix Equation of CNFDEs. Equation (5) with nonlinear terms $w(f) = f^2(t)$, $z(g) = g^2(t)$ are taken as an example in this section, while three linearized iterative methods are introduced. Based on the above, the linearized iterative schemes for CNFDEs are obtained, and the corresponding matrix equations are also derived. The CNFDEs under consideration are

$$\begin{cases} {}_0^C D_t^\alpha f(t) = f^2(t) + u_1(t, g(t)) \\ {}_0^C D_t^\beta g(t) = g^2(t) + u_2(t, f(t)) \end{cases}. \tag{6}$$

3.1. Direct linearized iterative scheme. Given the initial iteration values $\hat{f}$ and $\hat{g}$, substituting them into f^2 and g^2 yields the corresponding direct linearized equations:

$$\begin{cases} D_f^\alpha = \hat{f}^2(t) + u_1(t, g(t)) \\ D_g^\beta = \hat{g}^2(t) + u_2(t, f(t)) \end{cases}. \tag{7}$$

Applying the BLICM polynomial, the functions $f(t)$, $g(t)$ are approximately expressed as

$$f = \sum_{k=0}^n L_k(t) f_k, \quad g = \sum_{j=0}^n M_j(t) g_j, \tag{8}$$

where $f_k = f(t_k)$, $g_j = g(t_j)$, $L_k(t)$, $M_j(t)$ are interpolation basis functions:

$$L_k(t) = \frac{\frac{w_k}{t-t_k}}{\sum_{i=0, i \neq k}^n \frac{w_i}{t-t_i}}, \quad M_j(t) = \frac{\frac{w_j}{t-t_j}}{\sum_{l=0, l \neq j}^n \frac{w_l}{t-t_l}}. \tag{9}$$

And the first derivative of $L_k(t)$ can be expressed as

$$L'_k(t_i) = \begin{cases} \frac{w_k/w_i}{t_i - t_k}, & i \neq k \\ -\sum_{k=1, k \neq i}^n L'_k(t_i), & i = k \end{cases}. \quad (10)$$

Substitute the interpolation nodes into Equation (8). The matrix form of the d -th order derivative of Equation (8) is

$$\mathbf{f}^{(d)} = (D^{(d)}) \mathbf{f}, \quad (11)$$

where $\mathbf{f}^{(d)} = [f_0^{(d)}, \dots, f_n^{(d)}]^T$, the elements of $D^{(d)}$ are $L_k^{(d)}(t_i)$. A detailed introduction to the basis function can be found in [16]. Substituting Equation (8) into Equation (6), after rearrangement we obtain

$$\begin{aligned} \mathcal{G}_\alpha \sum_{k=0}^n [L'_k(0)t^{1-\alpha}] f_k + \mathcal{G}_\alpha \sum_{k=0}^n \left[\int_0^t \frac{L_k^{(2)}(\tau)}{(t-\tau)^{\alpha-1}} d\tau \right] f_k &= \hat{f}^2(t) + u_1(t, g(t)), \\ \mathcal{G}_\beta \sum_{j=0}^n [M'_j(0)t^{1-\beta}] g_j + \mathcal{G}_\beta \sum_{j=0}^n \left[\int_0^t \frac{M_j^{(2)}(\tau)}{(t-\tau)^{\beta-1}} d\tau \right] g_j &= \hat{g}^2(t) + u_2(t, f(t)). \end{aligned} \quad (12)$$

Let $t = t_\vartheta$, where $\vartheta = 0, 1, \dots, n$. Equation (12) can be written as

$$\begin{aligned} \mathcal{G}_\alpha \sum_{k=0}^n [L'_k(0)t_\vartheta^{1-\alpha}] f_k + \mathcal{G}_\alpha \sum_{k=0}^n \left[\int_0^{t_\vartheta} \frac{L_k^{(2)}(\tau)}{(t_\vartheta - \tau)^{\alpha-1}} d\tau \right] f_k &= \hat{f}^2(t_\vartheta) + u_1(t_\vartheta, g(t_\vartheta)), \\ \mathcal{G}_\beta \sum_{j=0}^n [M'_j(0)t_\vartheta^{1-\beta}] g_j + \mathcal{G}_\beta \sum_{j=0}^n \left[\int_0^{t_\vartheta} \frac{M_j^{(2)}(\tau)}{(t_\vartheta - \tau)^{\beta-1}} d\tau \right] g_j &= \hat{g}^2(t_\vartheta) + u_2(t_\vartheta, f(t_\vartheta)). \end{aligned} \quad (13)$$

For calculating the integral term in Equation (13), it is expressed using the Gauss quadrature formula with weight $\rho(\tau) = (t - \tau)^{1-\xi}$ as

$$\begin{aligned} Q_\gamma &= Q_\gamma(t_\vartheta) = \int_0^{t_\vartheta} \frac{L_k^{(2)}(\tau)}{(t_\vartheta - \tau)^{\alpha-1}} d\tau = \sum_{i=1}^h L_k^{(2)}(\tau_i^{\gamma, \xi}) A_i^{\gamma, \xi}, \\ Q_\ell &= Q_\ell(t_\vartheta) = \int_0^{t_\vartheta} \frac{M_j^{(2)}(\tau)}{(t_\vartheta - \tau)^{\beta-1}} d\tau = \sum_{i=1}^h M_j^{(2)}(\tau_i^{\ell, \xi}) A_i^{\ell, \xi}, \end{aligned} \quad (14)$$

where $A_i^{\gamma, \xi}$, $\tau_i^{\gamma, \xi}$ are Gaussian weights and Gaussian points, and h is the number of Gaussian points. Express Equation (6) in matrix form:

$$\begin{cases} \mathcal{G}_\alpha [T_\vartheta^\alpha D^{(1)} + Q_\gamma] F = \hat{F}^2 + U_1 \\ \mathcal{G}_\beta [T_\vartheta^\beta D^{(1)} + Q_\ell] G = \hat{G}^2 + U_2 \end{cases}, \quad (15)$$

where $D^{(1)}$ is derived by Equation (10), $U_1 = [u_1(t_0, g(t_0)), u_1(t_1, g(t_1)), \dots, u_1(t_n, g(t_n))]^T$, $\hat{F} = [\hat{f}_0, \hat{f}_1, \dots, \hat{f}_n]^T$, $U_2 = [u_2(t_0, f(t_0)), u_2(t_1, f(t_1)), \dots, u_2(t_n, f(t_n))]^T$, $\hat{G} = [\hat{g}_0, \hat{g}_1,$

$$\dots, \hat{g}_n]^T, F = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_n \end{bmatrix}, G = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_n \end{bmatrix}, T_\vartheta^\alpha = \begin{bmatrix} t_0^{1-\alpha} & & & \\ & t_1^{1-\alpha} & & \\ & & \ddots & \\ & & & t_n^{1-\alpha} \end{bmatrix}_{(n+1) \times (n+1)}.$$

We define the following transformation: $f \rightarrow f^{(\eta)}$, $g \rightarrow g^{(\eta)}$, $\hat{f} \rightarrow f^{(\eta-1)}$, $\hat{g} \rightarrow g^{(\eta-1)}$, η is the number of iterations. After the transformation, the iterative scheme of Equation (6) is

$$\begin{cases} D_{f^{(\eta)}}^\alpha = (f^{(\eta-1)}(t))^2 + u_1(t, g(t)) \\ D_{g^{(\eta)}}^\beta = (g^{(\eta-1)}(t))^2 + u_2(t, f(t)) \end{cases} \quad (16)$$

The corresponding matrix equations under the direct linearized iterative scheme are

$$\begin{cases} H_\gamma F^{(\eta)} = (F^{(\eta-1)})^2 + U_1 \\ H_\ell G^{(\eta)} = (G^{(\eta-1)})^2 + U_2 \end{cases} \quad (17)$$

where $H_\gamma = \mathcal{G}_\alpha [T_\gamma^\alpha D^{(1)} + Q_\gamma]$, $H_\ell = \mathcal{G}_\beta [T_\ell^\beta D^{(1)} + Q_\ell]$.

3.2. Partial linearized iterative scheme. Let one of the factors f , g in the nonlinear term f^2 , g^2 be unknown function, and substitute the initial iterative values into f^2 , g^2 , to get linear terms $\hat{f}f$, $\hat{g}g$:

$$\begin{cases} D_f^\alpha = \hat{f}f(t) + u_1(t, g(t)) \\ D_g^\beta = \hat{g}g(t) + u_2(t, f(t)) \end{cases} \quad (18)$$

We define the following transformation: $f \rightarrow f^{(\eta)}$, $g \rightarrow g^{(\eta)}$, $\hat{f} \rightarrow f^{(\eta-1)}$, $\hat{g} \rightarrow g^{(\eta-1)}$. After the transformation, the iterative scheme of Equation (5) is

$$\begin{cases} D_{f^{(\eta)}}^\alpha - f^{(\eta-1)}f^{(\eta)}(t) = u_1(t, g(t)) \\ D_{g^{(\eta)}}^\beta - g^{(\eta-1)}g^{(\eta)}(t) = u_2(t, f(t)) \end{cases} \quad (19)$$

The matrix equations under the partial linearized iterative scheme are

$$\begin{cases} (H_\gamma - \text{diag}(F^{(\eta-1)})) F^{(\eta)} = U_1 \\ (H_\ell - \text{diag}(G^{(\eta-1)})) G^{(\eta)} = U_2 \end{cases} \quad (20)$$

3.3. Newton linearized iterative scheme. Perform Taylor expansion on the f^2 , g^2 at the initial iterative values $\hat{f}$, $\hat{g}$, and only taking the linear parts, we have $f^2 \approx \hat{f}^2 + 2\hat{f}(f - \hat{f}) = 2\hat{f}f - \hat{f}^2$, $g^2 \approx \hat{g}^2 + 2\hat{g}(g - \hat{g}) = 2\hat{g}g - \hat{g}^2$. Through this process, the Newton linearized equation is

$$\begin{cases} D_f^\alpha = 2\hat{f}f(t) + u_1(t, g(t)) - \hat{f}^2 \\ D_g^\beta = 2\hat{g}g(t) + u_2(t, f(t)) - \hat{g}^2 \end{cases} \quad (21)$$

We define the following transformation: $f \rightarrow f^{(\eta)}$, $g \rightarrow g^{(\eta)}$, $\hat{f} \rightarrow f^{(\eta-1)}$, $\hat{g} \rightarrow g^{(\eta-1)}$. After the transformation, the iterative scheme of Equation (5) is

$$\begin{cases} D_{f^{(\eta)}}^\alpha - 2f^{(\eta-1)}f^{(\eta)}(t) = u_1(t, g(t)) - (f^{(\eta-1)})^2 \\ D_{g^{(\eta)}}^\beta - 2g^{(\eta-1)}g^{(\eta)}(t) = u_2(t, f(t)) - (g^{(\eta-1)})^2 \end{cases} \quad (22)$$

The matrix equations under the Newton linearized iterative scheme are

$$\begin{cases} (H_\gamma - 2\text{diag}(F^{(\eta-1)})) F^{(\eta)} = U_1 - (F^{(\eta-1)})^2 \\ (H_\ell - 2\text{diag}(G^{(\eta-1)})) G^{(\eta)} = U_2 - (G^{(\eta-1)})^2 \end{cases} \quad (23)$$

The discretization and implementation methods for the initial and boundary conditions of the CNFDEs are described in [21], and will not be repeated here. After applying the initial and boundary conditions, the final matrix equation is obtained. In the experiments,

we initialized the iteration counter at 1, set $\eta = 1$, the iteration accuracy $e = 10^{-8}$, and the maximum number of iterations to 200. The algorithm is stopped when the iteration limit of 200 is reached without convergence.

Define the vector $\mathbf{v} = (f_0(t_0), \dots, f_n(t_n), \dots, g_0(t_0), \dots, g_n(t_n))^T$. When the condition

$$\|\mathbf{v}^{(\eta)}(t) - \mathbf{v}^{(\eta-1)}(t)\|_2 = \sqrt{\sum_{k=0}^n (\mathbf{v}^{(\eta)}(t_k) - \mathbf{v}^{(\eta-1)}(t_k))^2} \leq e \quad (24)$$

is satisfied, the iteration terminates, and the numerical solution of the desired equation can then be obtained. In Equation (24), $\|\cdot\|_2$ is the Euclidean norm.

4. Numerical Experiments. This section will present three computational experiments aimed to demonstrate the accuracy and validity of the method. The absolute errors and relative errors of $f(t)$, $g(t)$ are defined as follows:

$$E_{af} = \|\tilde{\mathbf{f}} - \mathbf{f}\|_2, \quad E_{rf} = \frac{\|\tilde{\mathbf{f}} - \mathbf{f}\|_2}{\|\tilde{\mathbf{f}}\|_2},$$

$$E_{ag} = \|\tilde{\mathbf{g}} - \mathbf{g}\|_2, \quad E_{rg} = \frac{\|\tilde{\mathbf{g}} - \mathbf{g}\|_2}{\|\tilde{\mathbf{g}}\|_2},$$

where $\tilde{\mathbf{f}}$ and $\tilde{\mathbf{g}}$ represent the column vectors of the numerical solutions, while $\mathbf{f}$ and $\mathbf{g}$ denote the exact solutions.

4.1. Example 4.1. We consider the nonlinear coupled system as follows:

$$\begin{cases} {}_0^C D_t^\alpha f(t) = u_1(t) + f^2(t) + g^2(t), \\ {}_0^C D_t^\beta g(t) = u_2(t) + f^2(t) - g^2(t), & t \in (0, 1), \\ f(0) = f'(0) = 0, \quad g(0) = g'(0) = 0, \end{cases}$$

where $u_1(t) = \Gamma(2 + \alpha)t - (t^{1+\alpha})^2 - (t^{1+\frac{\beta}{2}})^2$, $u_2(t) = \frac{\Gamma(2+\frac{\beta}{2})}{\Gamma(2-\frac{\beta}{2})}t^{1-\frac{\beta}{2}} - (t^{1+\alpha})^2 + (t^{1+\frac{\beta}{2}})^2$.

The exact solutions are $f(t) = t^{1+\alpha}$, $g(t) = t^{1+\frac{\beta}{2}}$.

In Tables 1 to 3, the absolute errors, relative errors, and numbers of iterations under three iterative schemes in Chebyshev and equidistant nodes at $\alpha = \beta = 0.95$ are presented. From the results, it can be observed that the direct linearized iterative scheme requires

TABLE 1. Errors for Example 4.1, direct linearized iterative scheme

n	Chebyshev nodes					Equidistant nodes				
	E_{af}	E_{rf}	E_{ag}	E_{rg}	η	E_{af}	E_{rf}	E_{ag}	E_{rg}	η
3	1.7480×10^{-2}	1.5156×10^{-2}	3.4838×10^{-2}	2.8984×10^{-2}	27	4.3017×10^{-2}	3.8954×10^{-2}	5.6669×10^{-2}	4.8927×10^{-2}	23
5	3.2037×10^{-3}	2.3344×10^{-3}	8.1104×10^{-3}	5.6309×10^{-3}	14	1.4224×10^{-2}	1.1297×10^{-2}	2.1988×10^{-2}	1.6321×10^{-2}	15
8	6.7363×10^{-4}	4.0893×10^{-4}	2.4206×10^{-3}	1.3931×10^{-3}	13	6.0130×10^{-3}	4.0789×10^{-3}	1.0651×10^{-2}	6.6623×10^{-3}	14
10	3.4289×10^{-4}	1.8971×10^{-4}	1.3783×10^{-3}	7.2152×10^{-4}	12	4.1314×10^{-3}	2.5754×10^{-3}	7.6470×10^{-3}	4.3742×10^{-3}	13

TABLE 2. Errors for Example 4.1, partial linearized iterative scheme

n	Chebyshev nodes					Equidistant nodes				
	E_{af}	E_{rf}	E_{ag}	E_{rg}	η	E_{af}	E_{rf}	E_{ag}	E_{rg}	η
3	2.7435×10^{-2}	2.3788×10^{-2}	4.7178×10^{-2}	3.9250×10^{-2}	22	4.7710×10^{-2}	4.3203×10^{-2}	6.3896×10^{-2}	5.5167×10^{-2}	19
5	3.6031×10^{-3}	2.6254×10^{-3}	8.5818×10^{-3}	5.9582×10^{-3}	12	1.4269×10^{-2}	1.1333×10^{-2}	2.2065×10^{-2}	1.6378×10^{-2}	13
8	7.7244×10^{-4}	4.6891×10^{-4}	2.4972×10^{-3}	1.4371×10^{-3}	11	6.0142×10^{-3}	4.0797×10^{-3}	1.0653×10^{-2}	6.6634×10^{-3}	12
10	3.8639×10^{-4}	2.1377×10^{-4}	1.4079×10^{-3}	7.3703×10^{-4}	11	4.1317×10^{-3}	2.5756×10^{-3}	7.6475×10^{-3}	4.3745×10^{-3}	11

TABLE 3. Errors for Example 4.1, Newton linearized iterative scheme

n	Chebyshev nodes						Equidistant nodes					
	E_{af}	E_{rf}	E_{ag}	E_{rg}	η		E_{af}	E_{rf}	E_{ag}	E_{rg}	η	
3	3.8137×10^{-2}	3.3068×10^{-2}	6.5862×10^{-2}	5.4795×10^{-2}	22		5.1810×10^{-2}	4.6917×10^{-2}	7.3679×10^{-2}	6.3614×10^{-2}	19	
5	3.8601×10^{-3}	2.8126×10^{-3}	8.8282×10^{-3}	6.1293×10^{-3}	12		1.4322×10^{-2}	1.1376×10^{-2}	2.2091×10^{-2}	1.6397×10^{-2}	13	
8	8.6842×10^{-4}	5.2718×10^{-4}	2.5803×10^{-3}	1.4850×10^{-3}	11		6.0155×10^{-3}	4.0806×10^{-3}	1.0655×10^{-2}	6.6647×10^{-3}	11	
10	4.2848×10^{-4}	2.3706×10^{-4}	1.4395×10^{-3}	7.5353×10^{-4}	11		4.1320×10^{-3}	2.5758×10^{-3}	7.6481×10^{-3}	4.3748×10^{-3}	11	

the maximum number of iterations. The errors achieved by the three linearized iterative schemes are basically comparable, and the errors obtained with Chebyshev nodes are higher than with equidistant nodes. As the number of nodes increases, the errors obtained with both types of nodes gradually decrease.

In Table 4, the absolute errors obtained by BLICM with Chebyshev nodes under three iterative schemes at $n = 5, 8, 10$, $\alpha = \beta = 0.75$ are shown. From the comparison, it can be seen that with the increase number of interpolation node, the absolute errors gradually decrease. Under the same number of interpolation nodes, the smaller absolute error is achieved by direct linearized iterative scheme. Compared with the numerical results obtained by the Legendre spectral collocation method in [22], it can be seen that the BLICM can achieve higher accuracy with fewer interpolation nodes.

TABLE 4. Errors for different iterative schemes $\alpha = \beta = 0.75$

$\alpha = \beta = 0.75$	Chebyshev nodes						[22]		
	Direct linearized		Partial linearized		Newton linearized		Legendre spectral collocation method		
	E_{af}	E_{ag}	E_{af}	E_{ag}	E_{af}	E_{ag}	Δt	E_{af}	E_{ag}
$n = 5$	8.6741×10^{-3}	1.2449×10^{-2}	1.0019×10^{-2}	1.3480×10^{-2}	1.4349×10^{-2}	1.5165×10^{-2}	1/40	5.34×10^{-2}	1.35×10^{-2}
$n = 8$	1.7501×10^{-3}	3.9159×10^{-3}	2.0607×10^{-3}	4.0569×10^{-3}	2.3393×10^{-3}	4.2154×10^{-3}	1/160	1.41×10^{-2}	3.50×10^{-3}
$n = 10$	8.4796×10^{-4}	2.2690×10^{-3}	9.9307×10^{-4}	1.0956×10^{-3}	1.1315×10^{-3}	2.3797×10^{-3}	1/640	3.60×10^{-3}	9.00×10^{-4}

In Table 5, the relative errors obtained by two kinds of nodes under Newton linearized iterative scheme are presented, at different values of n and $\alpha = \beta$. For fixed $\alpha = \beta$, as n increases from 6 to 9, the errors for $f(t)$ and $g(t)$ obtained by Chebyshev nodes decrease from 10^{-3} to 10^{-4} . The errors for $f(t)$ and $g(t)$ obtained by equidistant nodes decrease from 10^{-2} to 10^{-3} , while those for $f(t)$ remain around 10^{-3} at $\alpha = \beta = 0.99$. For different values of $\alpha = \beta$, the errors all decrease with the increase values of n .

TABLE 5. Relative errors for Newton linearized iterative scheme, $\alpha = \beta$

$\alpha = \beta$	Chebyshev nodes				Equidistant nodes			
	$n = 6$		$n = 9$		$n = 6$		$n = 9$	
	E_{rf}	E_{rg}	E_{rf}	E_{rg}	E_{rf}	E_{rg}	E_{rf}	E_{rg}
$\alpha = \beta = 0.65$	5.8511×10^{-3}	7.1678×10^{-3}	1.6025×10^{-3}	2.0752×10^{-3}	1.6612×10^{-2}	1.1705×10^{-2}	5.8053×10^{-3}	4.2675×10^{-3}
$\alpha = \beta = 0.77$	3.3552×10^{-3}	5.4906×10^{-3}	8.0441×10^{-4}	1.5809×10^{-3}	1.1561×10^{-2}	1.1088×10^{-2}	4.3678×10^{-3}	4.5331×10^{-3}
$\alpha = \beta = 0.82$	2.6536×10^{-3}	4.9060×10^{-3}	6.1514×10^{-4}	1.4093×10^{-3}	1.0214×10^{-2}	1.1137×10^{-2}	3.9775×10^{-3}	4.7094×10^{-3}
$\alpha = \beta = 0.99$	1.3528×10^{-3}	3.3618×10^{-3}	3.0933×10^{-4}	9.3230×10^{-4}	6.9091×10^{-3}	1.1934×10^{-2}	2.9873×10^{-3}	5.4683×10^{-3}

In Figure 1, the comparisons between the numerical solutions and the analytical solutions are presented. From the figure, it can be observed that the numerical solutions numerically obtained by BLICM are in high agreement with the analytical solutions. The numerical solutions of Example 4.1 obtained by Newton linearized iterative scheme at different values of $\alpha = \beta$ are shown in Figure 2. The numerical solutions of Example 4.1 obtained by Newton linearized iterative scheme with fixed α or β are shown in Figure 3. It can be seen from Figure 3 that both $f(t)$ and $g(t)$ are monotonically increasing in $t \in [0, 1]$, and their growth rates accelerate with the increase of α and β .

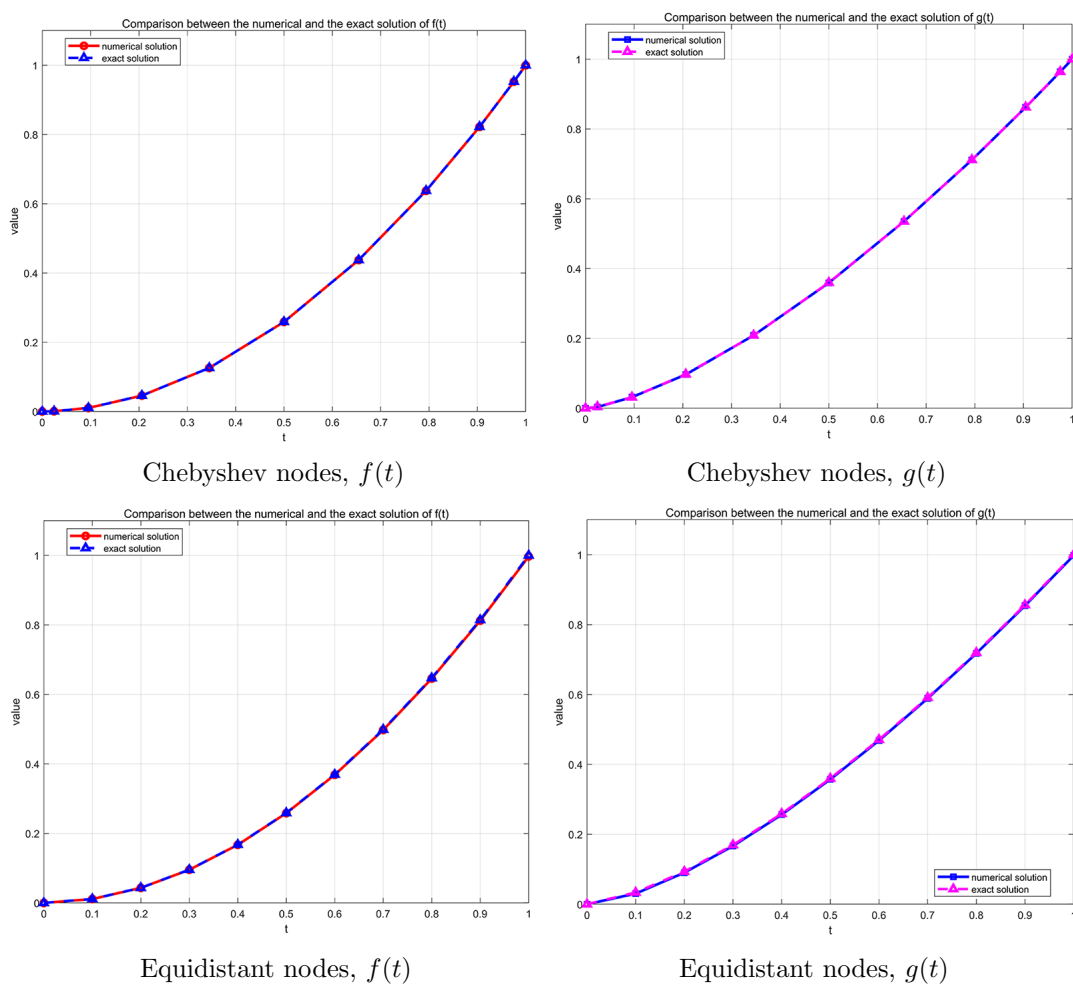
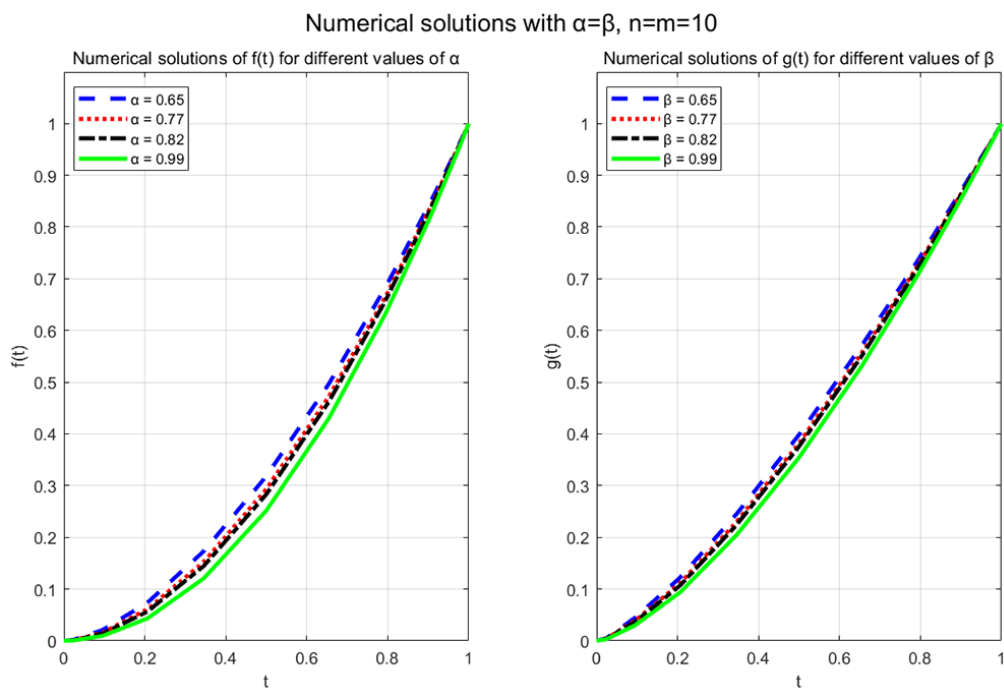
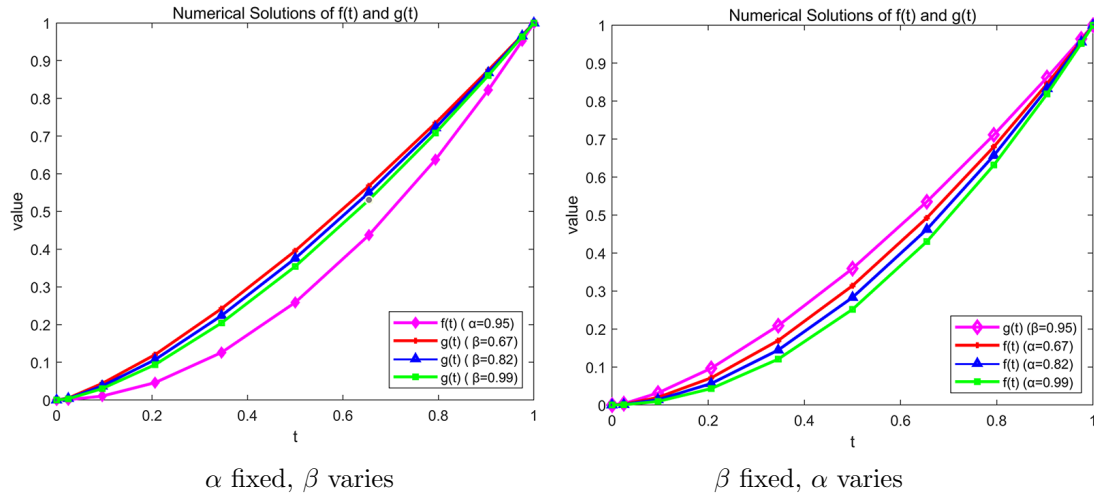


FIGURE 1. Comparisons of numerical solutions and exact solutions

FIGURE 2. $n = 10$, the numerical results with $\alpha = \beta$


 FIGURE 3. $\alpha = 0.95$ and $\beta = 0.95$

In Table 6, the relative errors obtained by Chebyshev nodes under three iterative schemes are presented at $\alpha = 0.95$ ($\beta = 0.67, 0.82, 0.99$) and $\beta = 0.95$ ($\alpha = 0.67, 0.82, 0.99$). From the result comparison, it can be observed that for the fixed value of α , as the value of β gradually increases, the errors for $f(t)$ and $g(t)$ gradually decrease; similarly, for the fixed value of β , as the value of α gradually increases, the errors for $f(t)$ and $g(t)$ also gradually decrease.

 TABLE 6. Relative errors with α fixed or β fixed under different linearized iterative schemes

$\alpha \neq \beta$	$\alpha = 0.95$							
	$\beta = 0.67$		$\beta = 0.82$		$\beta = 0.99$		$\beta = 0.95$	
	E_{rf}	E_{rg}	E_{rf}	E_{rg}	E_{rf}	E_{rg}	E_{rf}	E_{rg}
Direct linearized iterative scheme	2.6095×10^{-4}	1.3566×10^{-3}	2.0338×10^{-4}	9.9218×10^{-4}	1.9727×10^{-4}	6.5343×10^{-4}	5.8381×10^{-4}	7.4831×10^{-4}
Partial linearized iterative scheme	3.1918×10^{-4}	1.3809×10^{-3}	2.4564×10^{-4}	1.0693×10^{-3}	2.1645×10^{-4}	6.6759×10^{-4}	2.9221×10^{-4}	1.3789×10^{-3}
Newton linearized iterative scheme	3.7457×10^{-4}	1.4105×10^{-3}	2.8607×10^{-4}	1.0351×10^{-3}	6.8236×10^{-4}	8.5708×10^{-4}	2.5230×10^{-4}	7.5209×10^{-4}
							2.9017×10^{-4}	7.2757×10^{-4}
							1.7225×10^{-4}	7.2050×10^{-4}
							2.2415×10^{-4}	1.0109×10^{-3}
							1.9450×10^{-4}	7.3580×10^{-4}
							2.1603×10^{-4}	7.5209×10^{-4}

4.2. **Example 4.2.** We consider the nonlinear coupled system as follows:

$$\begin{cases} {}^C_0 D_t^\alpha f(t) = u_1(t) + \sin(f(t)) + g^2(t), \\ {}^C_0 D_t^\beta g(t) = u_2(t) + \sin(g(t)) + f^2(t), \quad t \in (0, 1), \\ f(0) = f'(0) = 0, g(0) = g'(0) = 0, \end{cases}$$

where $u_1(t) = \frac{\Gamma(5+\alpha)}{\Gamma(5)} t^4 + \sin(t)^{4+\alpha} - t^{10+2\beta}$, $u_2(t) = \frac{\Gamma(6+\beta)}{\Gamma(6)} t^5 + \sin(t)^{5+\beta} - t^{8+2\alpha}$. Its exact solutions are $f(t) = t^{4+\alpha}$, $g(t) = t^{5+\beta}$.

In Tables 7 and 8, the absolute errors, relative errors, and numbers of iterations under two iterative schemes in Chebyshev and equidistant nodes at $\alpha = \beta = 0.95$ are presented. From the results, it can be observed that the direct linearized iterative scheme requires

TABLE 7. Errors for Example 4.2, direct linearized iterative scheme

n	Chebyshev nodes					Equidistant nodes				
	E_{af}	E_{rf}	E_{ag}	E_{rg}	η	E_{af}	E_{rf}	E_{ag}	E_{rg}	η
5	6.0822×10^{-4}	5.1512×10^{-4}	2.4182×10^{-3}	2.4182×10^{-3}	21	8.0457×10^{-4}	7.5957×10^{-3}	3.9458×10^{-3}	3.8038×10^{-3}	21
7	1.5958×10^{-6}	1.2057×10^{-6}	1.5546×10^{-6}	1.2131×10^{-6}	18	7.8063×10^{-6}	6.9349×10^{-6}	6.1919×10^{-6}	5.6864×10^{-6}	18
9	1.0809×10^{-7}	7.4424×10^{-8}	4.3027×10^{-8}	3.0665×10^{-8}	16	1.0785×10^{-6}	9.0198×10^{-7}	3.4386×10^{-7}	2.9989×10^{-7}	17
11	1.3164×10^{-8}	8.3809×10^{-9}	8.8224×10^{-9}	5.8236×10^{-9}	14	2.4047×10^{-7}	1.9006×10^{-7}	4.8201×10^{-8}	3.9981×10^{-8}	15

TABLE 8. Errors for Example 4.2, Newton linearized iterative scheme

n	Chebyshev nodes					Equidistant nodes				
	E_{af}	E_{rf}	E_{ag}	E_{rg}	η	E_{af}	E_{rf}	E_{ag}	E_{rg}	η
5	1.4624×10^{-3}	1.2386×10^{-3}	6.5547×10^{-3}	5.7136×10^{-3}	15	7.9773×10^{-4}	7.5312×10^{-4}	3.9558×10^{-3}	3.8134×10^{-3}	13
7	1.4662×10^{-5}	1.1078×10^{-5}	1.3084×10^{-5}	1.0210×10^{-5}	14	8.0235×10^{-6}	7.1279×10^{-6}	6.3645×10^{-6}	5.8449×10^{-6}	12
9	1.6361×10^{-6}	1.1265×10^{-6}	4.7272×10^{-7}	3.3691×10^{-7}	12	1.0814×10^{-6}	9.0440×10^{-7}	3.4457×10^{-7}	3.0051×10^{-7}	11
11	2.8881×10^{-7}	1.8387×10^{-7}	5.2933×10^{-8}	3.4940×10^{-8}	12	2.4799×10^{-7}	1.9600×10^{-7}	4.9777×10^{-8}	4.1287×10^{-8}	10

the maximum number of iterations. For the same numbers of n , the errors obtained by Chebyshev nodes are higher than equidistant nodes. As the number of nodes increases, the errors obtained by two kinds of nodes gradually improve. Since the nonlinear term is $\sin(u(t))$, $u(t)$ cannot only be effectively treated by partial linearized iterative scheme. Therefore, only the E_a and E_r for $f(t)$ and $g(t)$ under the direct and Newton linearized iterative schemes are presented here.

In Table 9, the relative errors obtained by two kinds of nodes under two linearized iterative schemes are presented at different values of $\alpha = \beta$ in $n = 10$. From the result comparison, it can be observed that for the same values of $\alpha = \beta$, the errors for $f(t)$ obtained by Chebyshev nodes are 10^{-7} and 10^{-8} , and the errors for $g(t)$ are also 10^{-7} and 10^{-8} ; the errors for $f(t)$ and $g(t)$ obtained by equidistant nodes are 10^{-7} under different iterative schemes. When the values of $\alpha = \beta$ are different, the errors obtained by two kinds of nodes all decrease with the increase value of $\alpha = \beta$ under the same iteration scheme.

TABLE 9. Relative errors obtained by different nodes under linearized iterative schemes, $\alpha = \beta$, $n = 10$

$\alpha = \beta$	Direct linearized iterative scheme				Partial linearized iterative scheme			
	Chebyshev nodes		Equidistant nodes		Chebyshev nodes		Equidistant nodes	
	E_{rf}	E_{rg}	E_{rf}	E_{rg}	E_{rf}	E_{rg}	E_{rf}	E_{rg}
0.7	8.4094×10^{-8}	3.0156×10^{-8}	5.9880×10^{-7}	1.4845×10^{-7}	9.7301×10^{-7}	2.8084×10^{-7}	5.9317×10^{-7}	1.4250×10^{-7}
0.8	4.7869×10^{-8}	1.5204×10^{-8}	5.1869×10^{-7}	1.1902×10^{-7}	6.1722×10^{-7}	1.4126×10^{-7}	5.0717×10^{-7}	1.1234×10^{-7}
0.9	2.1896×10^{-8}	9.6702×10^{-9}	3.3149×10^{-7}	7.7057×10^{-8}	2.7039×10^{-7}	6.0940×10^{-8}	3.3003×10^{-7}	7.7664×10^{-8}

4.3. **Example 4.3.** We consider the nonlinear coupled system as follows:

$$\begin{cases} {}^C_0 D_t^\alpha f(t) = u_1(t) - f^3(t) - g^3(t), \\ {}^C_0 D_t^\beta g(t) = u_2(t) + f^3(t) - g^3(t), \\ f(0) = f'(0) = 0, g(0) = g'(0) = 0, \end{cases} \quad t \in (0, 1),$$

where $u_1(t) = \frac{\Gamma(5+\alpha)}{\Gamma(5)}t^4 + t^{12+3\alpha} + t^{15+3\beta}$, $u_2(t) = \frac{\Gamma(6+\beta)}{\Gamma(6)}t^5 - t^{12+3\alpha} + t^{15+3\beta}$. Its exact solutions are $f(t) = t^{4+\alpha}$, $g(t) = t^{5+\beta}$.

In Tables 10 to 12, the absolute errors, relative errors, and numbers of iterations under three iterative schemes in Chebyshev and equidistant nodes at $\alpha = \beta = 0.85$ are presented. From the results, it can be observed that the direct linearized iterative scheme requires the

TABLE 10. Errors for Example 4.3, direct linearized iterative scheme

n	Chebyshev nodes					Equidistant nodes				
	E_{af}	E_{rf}	E_{ag}	E_{rg}	η	E_{af}	E_{rf}	E_{ag}	E_{rg}	η
5	1.3547×10^{-4}	1.1474×10^{-4}	1.0431×10^{-3}	9.0927×10^{-4}	16	2.1049×10^{-4}	1.9872×10^{-4}	2.1492×10^{-3}	2.0719×10^{-3}	20
7	1.0451×10^{-6}	7.8961×10^{-7}	8.9966×10^{-7}	7.0202×10^{-7}	13	4.1105×10^{-6}	3.6517×10^{-6}	3.4896×10^{-6}	3.2047×10^{-6}	15
9	8.0848×10^{-8}	5.5665×10^{-8}	3.1840×10^{-8}	2.2692×10^{-8}	11	5.1908×10^{-7}	4.3412×10^{-7}	2.0226×10^{-7}	1.7640×10^{-7}	13
11	1.6584×10^{-8}	1.0558×10^{-8}	6.7310×10^{-9}	4.4430×10^{-9}	10	1.1894×10^{-7}	9.4003×10^{-8}	2.7770×10^{-8}	2.3034×10^{-8}	12

TABLE 11. Errors for Example 4.3, partial linearized iterative scheme

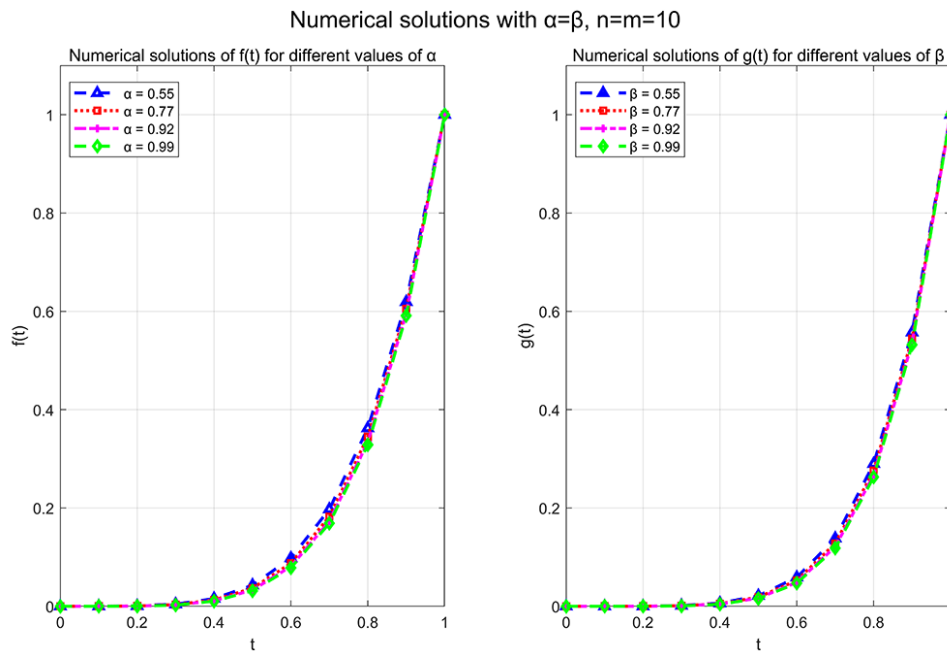
n	Chebyshev nodes					Equidistant nodes				
	E_{af}	E_{rf}	E_{ag}	E_{rg}	η	E_{af}	E_{rf}	E_{ag}	E_{rg}	η
5	2.0439×10^{-4}	1.7310×10^{-4}	1.3980×10^{-3}	1.2186×10^{-3}	13	2.2870×10^{-4}	2.1591×10^{-4}	1.8186×10^{-3}	1.7532×10^{-3}	16
7	1.1478×10^{-6}	8.6718×10^{-7}	7.8571×10^{-7}	6.1310×10^{-7}	11	4.1113×10^{-6}	3.6524×10^{-6}	3.4683×10^{-6}	3.1852×10^{-6}	13
9	8.4508×10^{-8}	5.8185×10^{-8}	3.1834×10^{-8}	2.2688×10^{-8}	10	5.1948×10^{-7}	4.3445×10^{-7}	2.0282×10^{-7}	1.7689×10^{-7}	12
11	1.1941×10^{-8}	7.6023×10^{-9}	2.5957×10^{-9}	1.7134×10^{-9}	10	1.1879×10^{-8}	9.3884×10^{-8}	2.7952×10^{-8}	2.3184×10^{-8}	11

TABLE 12. Errors for Example 4.3, Newton linearized iterative scheme

n	Chebyshev nodes					Equidistant nodes				
	E_{af}	E_{rf}	E_{ag}	E_{rg}	η	E_{af}	E_{rf}	E_{ag}	E_{rg}	η
5	3.3848×10^{-4}	2.8666×10^{-4}	2.5447×10^{-3}	2.2182×10^{-3}	11	2.7129×10^{-4}	2.5611×10^{-4}	1.6282×10^{-3}	1.5696×10^{-3}	13
7	2.7206×10^{-6}	2.0555×10^{-6}	2.1055×10^{-6}	1.6429×10^{-6}	11	4.1936×10^{-6}	3.7255×10^{-6}	3.5066×10^{-6}	3.2203×10^{-6}	11
9	1.0629×10^{-7}	7.3180×10^{-8}	5.3172×10^{-8}	6.7896×10^{-8}	10	5.2010×10^{-7}	4.3497×10^{-7}	2.0180×10^{-7}	1.7600×10^{-7}	10
11	1.2006×10^{-8}	7.6439×10^{-9}	2.7257×10^{-9}	1.7992×10^{-9}	10	1.1905×10^{-7}	9.4092×10^{-8}	2.9472×10^{-8}	2.4446×10^{-8}	10

largest number of iterations. For the same numbers of n , the errors achieved by Chebyshev nodes are higher than the equidistant nodes. As the number of nodes increases, the errors obtained by both kinds of nodes gradually improve.

The numerical solutions of Example 4.3 obtained by Newton linearized iterative scheme at different values of $\alpha = \beta$ are shown in Figure 4. Both solutions are monotonically increasing in $t \in [0, 1]$, growing slowly at the initial stage and accelerating markedly with time, which matches the non-local memory effect of fractional-order systems.


 FIGURE 4. Numerical solutions with $\alpha = \beta$, $n = 10$

In Table 13, the relative errors obtained by Chebyshev nodes and equidistant nodes under Newton linearized iterative scheme are presented, at different values of n and $\alpha = \beta$. From the comparison of results, it can be observed that for a fixed value of $\alpha = \beta$, when the number of nodes increases from $n = 6$ to $n = 9$, the accuracy of the numerical solutions for $f(t)$ and $g(t)$ using Chebyshev nodes is improved by 10^2 . Meanwhile, the accuracy obtained with equidistant nodes is also enhanced by 10^2 . For different values of $\alpha = \beta$, all numerical errors decrease as n increases.

TABLE 13. Relative errors for Newton linearized iterative scheme, with different $\alpha = \beta$

$\alpha = \beta$	Chebyshev nodes				Equidistant nodes			
	$n = 6$		$n = 9$		$n = 6$		$n = 9$	
	E_{rf}	E_{rg}	E_{rf}	E_{rg}	E_{rf}	E_{rg}	E_{rf}	E_{rg}
$\alpha = \beta = 0.55$	8.2022×10^{-6}	1.1882×10^{-5}	1.6078×10^{-7}	6.1694×10^{-8}	2.0267×10^{-5}	2.2585×10^{-5}	4.5035×10^{-7}	1.3829×10^{-7}
$\alpha = \beta = 0.77$	5.2103×10^{-6}	9.6841×10^{-6}	1.0199×10^{-7}	4.9672×10^{-8}	2.2562×10^{-5}	3.3788×10^{-5}	4.9824×10^{-7}	1.9412×10^{-7}
$\alpha = \beta = 0.92$	2.2790×10^{-6}	1.1746×10^{-5}	4.1070×10^{-8}	2.3339×10^{-8}	1.3841×10^{-5}	2.5917×10^{-5}	2.9973×10^{-7}	1.2571×10^{-7}
$\alpha = \beta = 0.99$	3.6848×10^{-7}	2.7818×10^{-6}	9.1683×10^{-9}	4.0510×10^{-9}	2.2838×10^{-6}	4.7447×10^{-6}	4.8264×10^{-8}	2.1506×10^{-8}

5. Concluding Remarks. The BLICM to solve CNFDEs is proposed in this paper. Three linearized iterative schemes for the equations are constructed. Integration by parts is employed to remove the singularity in the Caputo fractional derivatives, thereby transforming them into Riemann integrals. The BLICM is then used to discretize the equations, obtaining matrix equations corresponding to each iterative scheme. It is shown by the numerical examples that the accuracy achieved with Chebyshev nodes is higher than with equidistant nodes, and the fewest iterations are required by the Newton linearized iterative scheme in all three cases. It is indicated by the experimental results that high computational accuracy is achieved by the BLICM in solving the CNFDEs, thereby verifying the effectiveness of method. Some limitations should be noted that a rigorous convergence and stability analysis of the proposed method has not yet been carried out, and its theoretical properties remain to be further investigated. The present numerical experiments mainly verify the effectiveness of the method from a computational perspective; it can be observed that the error gradually decreases as the number of nodes increases. In future work, we will evaluate the convergence rate of the error as the step size decreases. The fractional-order term in the fractional differential equation can be generalized to variable-order time fractional term. The variable-order time-fractional differential equation extends the originally constant order α to a time-dependent function $\alpha(t)$, enabling the description of time-varying and non-stationary systems. The problems are rendered more difficult to solve numerically, and direct time discretization schemes require large memory and high computational cost. Therefore, in the future, we will study the numerical solution of variable-order time-fractional differential equations using the barycentric interpolation collocation method combined with iterative schemes.

Acknowledgment. This work was supported by National Natural Science Foundation of Shandong Province (Grant No. ZR2022MA003), Doctoral Research Fund of Shandong Jiaozhu University (Grant No. X25108).

REFERENCES

- [1] A. S. Deogan, R. Dilz and D. Caratelli, On the application of fractional derivative operator theory to the electromagnetic modeling of frequency dispersive media, *Mathematics*, vol.12, no.7, 932, 2024.
- [2] J. R. Padhi, M. A. Deeb, S. Tripathy and M. K. Debnath, Fractional calculus based PI-FOPID controller for frequency deviation control in integrated power system, in *Control Applications in Modern Power Systems. Lecture Notes in Electrical Engineering*, J. Kumar, M. Tripathy and P. Jena (eds.), Singapore, Springer, 2023.
- [3] M. Partohaghghi, M. Mortezaee and A. Akgül, Numerical analysis of the fractal-fractional diffusion model of ignition in the combustion process, *Alexandria Engineering Journal*, vol.86, pp.1-8, 2024.
- [4] Y. Tang, P. Wang, P. Ren and H. Zhang, Fractional derivative-based normalized viscoelastic model of strain-hardening clays, *Frontiers in Materials*, vol.11, 1338251, 2024.
- [5] S. B. Yuste, Weighted average finite difference methods for fractional diffusion equations, *Journal of Computational Physics*, vol.216, no.1, pp.264-274, 2006.

- [6] S. U. Rahman and J. L. D. Palencia, Analytical and numerical conceptions of flame propagation in porous media with a generalization in diffusion and reaction modeling, *Combustion Science and Technology*, vol.197, no.17, pp.4602-4625, 2025.
- [7] Z. Sheng, Y. Liu and Y. Li, Finite element method combined with time graded meshes for the time-fractional coupled Burgers' equations, *Journal of Applied Mathematics and Computing*, vol.70, no.1, pp.513-533, 2024.
- [8] P. Assari and M. Dehghan, A meshless local Galerkin method for solving Volterra integral equations deduced from nonlinear fractional differential equations using the moving least squares technique, *Applied Numerical Mathematics*, vol.143, pp.276-299, 2019.
- [9] M. Subramanian, M. Manigandan, C. Tunç et al., On system of nonlinear coupled differential equations and inclusions involving Caputo-type sequential derivatives of fractional order, *Journal of Taibah University for Science*, vol.16, no.1, pp.1-23, 2022.
- [10] M. L. Maheswari, K. S. Shri and M. Sajid, Analysis on existence of system of coupled multifractional nonlinear hybrid differential equations with coupled boundary conditions, *AIMS Mathematics*, vol.9, pp.13642-13658, 2024.
- [11] T. Nabil, Ulam stabilities of nonlinear coupled system of fractional differential equations including generalized Caputo fractional derivative, *AIMS Mathematics*, vol.6, pp.5088-5105, 2021.
- [12] Y. Chen, X. Ke, X. Han, Y. Sun and L. Liu, Wavelet method for solving fractional differential equations and analysis of its convergence, *Journal of Shandong University (Natural Science Edition)*, vol.50, no.2, pp.67-74, 2015.
- [13] J. Xie, T. Wang, Z. Ren, J. Zhang and L. Quan, Haar wavelet method for approximating the solution of a coupled system of fractional-order integral-differential equations, *Mathematics and Computers in Simulation*, vol.163, pp.80-89, 2019.
- [14] M. A. Zaky and I. G. Ameen, A priori error estimates of a Jacobi spectral method for nonlinear systems of fractional boundary value problems and related Volterra-Fredholm integral equations with smooth solutions, *Numerical Algorithms*, vol.84, no.1, pp.63-89, 2020.
- [15] J. P. Berrut, R. Baltensperger and H. D. Mittelmann, Recent developments in barycentric rational interpolation, *Trends and Applications in Constructive Approximation*, pp.27-51, 2005.
- [16] J. Li and B. Ding, Barycentric Lagrange interpolation collocation method for solving nonlinear fractional pseudo-parabolic equations, *Journal of Applied Mathematics and Computing*, vol.72, Article no. 9494, 2026.
- [17] 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.
- [18] X. Zhang, Y. Chen and L. Wei, High-order numerical approximation for 2D time-fractional advection-diffusion equation under Caputo derivative, *Fractal & Fractional*, vol.8, no.8, 474, 2024.
- [19] J. Li and J. Qu, High precision barycentric interpolation collocation method to solve EFK equation, *International Journal of Computer Mathematics*, vol.102, no.11, pp.1-19, 2025.
- [20] J. Li, X. Su and K. Zhao, Barycentric interpolation collocation algorithm to solve fractional differential equations, *Mathematics and Computers in Simulation*, vol.205, pp.340-367, 2023.
- [21] J. Li and J. Qu, Barycentric lagrange interpolation collocation method for solving the Sine-Gordon equation, *Wave Motion*, vol.120, 103159, 2023.
- [22] D. Yue, *Numerical Methods and Derivative Order Estimation for Coupled Fractional Differential Equations*, Ph.D. Thesis, Guizhou Normal University, 2022.

Author Biography



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, China, and worked as postdoctoral from April 2013 to April 2017.

Dr. Li is currently a professor at the School of Science, and he works at the Computational Intelligence Center, 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, 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.



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 at Shandong Jianzhu University. Her primary research interests are theory of partial differential equations and barycentric interpolation collocation method.



Xiaohan Yu received her bachelor's degree from the School of Statistics and Mathematics, Shandong University of Finance and Economics, China, in 2023. She is currently a master's student in the School of Science at Shandong Jianzhu University, China. Her primary research interest is machine learning.



Yanlu Liu received her bachelor's degree from the School of Science, Shandong Jianzhu University, China, in 2025. She is currently a master's student in the School of Science at Shandong Jianzhu University. Her primary research interests are machine learning and the barycentric interpolation collocation method.



Xu Yuan received his bachelor's degree from the School of Data Science, Qufu Normal University, China, in 2024. He is currently a master's student in the School of Science at Shandong Jianzhu University, China. His primary research interests are applied statistics and data analysis.