Computational and Applied Mathematics (2024) 43:68
https://doi.org/10.1007/s40314-023-02573-7

An efficient iterative method for multi-order nonlinear fractional differential equations based on the integrated Bernoulli polynomials

Babak Azarnavid[1] · Mahdi Emamjomeh[2] · Mohammad Nabati[3] · Abdollah Dinmohammadi[4]

Received: 3 September 2023 / Revised: 1 December 2023 / Accepted: 9 December 2023

Abstract

We present an effective practical approach for solving multi-order nonlinear fractional differential equations. Our method uses integrated Bernoulli polynomials and comes with a comprehensive convergence analysis. The integrated Bernoulli polynomials are combined with the collocation and simple iteration methods to approximate the solutions. We have provided several numerical examples to demonstrate the effectiveness, strength, and flexibility of our method. The results obtained from implementing the method have been compared with exact solutions and results obtained from other methods mentioned in the articles.

Keywords Multi-order nonlinear fractional differential equation · Integrated Bernoulli polynomials · Iterative method · Convergence analysis

Mathematics Subject Classification 65J15 · 26A33 · 34L30 · 33F05 · 41A10

1 Introduction

Over the past few decades, there has been significant research into fractional order differential equations due to their effectiveness in describing various real-world problems. When it comes to complex phenomena in different scientific fields, fractional calculus, and fractional order

Communicated by Agnieszka Malinowska.

✉ Babak Azarnavid
babakazarnavid@ubonab.ac.ir

[1] Department of Mathematics, Basic Science Faculty, University of Bonab, Bonab, Iran

[2] Basic Sciences Group, Golpayegan College of Engineering, Isfahan University of Technology, Golpayegan 87717-67498, Iran

[3] Department of Basic Sciences, Abadan Faculty of Petroleum Engineering, Petroleum University of Technology, Abadan, Iran

[4] Department of Mathematics, Buein Zahra Technical University, Buein Zahra, Qazvin, Iran

derivatives offer a more accurate description than integer order derivatives. This is particularly true when dealing with memory effects. Fractional differential equations are utilized to model practical problems in various fields. As an example, refer to (Prakash et al. 2017; Chen et al. 2022; Shi et al. 2022; Alqhtani et al. 2022; Odibat and Baleanu 2022), as well as the references mentioned therein. Researchers have presented several numerical methods for solving fractional differential equations (Bakhshandeh-Chamazkoti and Alipour 2022; Alipour and Agahi 2018; Azarnavid et al. 2022; Rezabeyk et al. 2023; Khodabandelo et al. 2022; Eftekhari and Rashidinia 2023). On the other hand, there is a significant category of fractional differential equations that involve various orders of fractional derivatives of the desired function. These types of equations are called multi-term or multi-order fractional differential equations. They have appeared in various fields, such as mechanics Joujehi et al. (2022), chemical reactor theory Erturk et al. (2022), and visco-elastic damping Dadkhah et al. (2020). It is often impossible to determine the analytical solutions for multi-order problems. Hence, we require robust numerical methods to solve them accurately. Researchers have offered several methods and techniques to address this particular set of problems. For example, refer to the Amin et al. (2022), Abd-Elhameed and Alsuyuti (2023), Nagy (2022) and the related references mentioned in them.

This paper studies the following general form of nonlinear multi-order fractional initial value problems:

$$\begin{cases} \partial^{\alpha} u(t) - \mathcal{G}(t, u(t), \partial^{\beta_1} u(t), \partial^{\beta_2} u(t), ..., \partial^{\beta_m} u(t)) = 0, & t \in [0, 1], \\ \frac{d^i u(t)}{dt^i}|_{t=0} = u_i, \quad i = 0, ..., z-1, \quad z \in N, \end{cases} \tag{1}$$

where $\alpha \in (z-1, z], \alpha > \beta_m > ... > \beta_1 > 0$, $\mathcal{G}$ has enough smoothness to guarantee the existence and uniqueness of the solution and all derivatives are in Caputo sense. Several research works, such as (Verma and Kumar 2022; Diethelm and Ford 2004; Diethelm 2010), have investigated the existence and uniqueness of the solution of these classes of equations. As far as we know, in comparison to other types of fractional equations, few articles have dealt with their numerical simulation. Therefore, we have been interested in providing a practical and efficient numerical method to solve such problems.

As we know, in recent years, methods based on Bernoulli polynomials have been used to solve different differential equations (Bhrawy et al. 2012; Azarnavid 2023; Postavaru and Toma 2022; Postavaru 2022). Bernoulli polynomials have advantages over some classical orthogonal polynomials for approximating an arbitrary unknown function. Some of them are mentioned in Bhrawy et al. (2012). Here, we propose a vigorous method based on the integrated Bernoulli polynomials combined with the collocation and simple iteration methods to approximate the solutions. As far as we have checked, this is the first try at utilizing integrated Bernoulli polynomials for differential equations. We have also given a rigorous convergence analysis of the proposed iterative method for nonlinear problems. Finally, several numerical examples of the implementation of the method and comparisons are presented to test the method's efficiency, power, and versatility.

2 Basic definitions and theorems

This section will present the definitions, theorems, and results regarding fractional calculations and Bernoulli polynomials.

Definition 2.1 For any $t > 0$, the Riemann–Liouville fractional integral operator and Caputo's fractional derivative operator of arbitrary order $\alpha > 0$ for a function $\mathcal{H}$ can be defined as follows, respectively:

$$\mathcal{J}^{\alpha}\mathcal{H}(t) = \frac{1}{\Gamma(\alpha)}\int_0^t (t-s)^{\alpha-1}\mathcal{H}(s)ds, \tag{2}$$

and

$$\partial^{\alpha}\mathcal{H}(t) = \frac{1}{\Gamma(z-\alpha)}\int_0^t (t-s)^{z-\alpha-1}\mathcal{H}^{(z)}(s)ds, \qquad z-1<\alpha\le z, z\in N. \tag{3}$$

For $\alpha > 0, \beta > 0, z-1 < \alpha \le z, z \in N$ and $t > 0$, they have the following properties Diethelm (2010):

$$\begin{aligned}
&\mathcal{J}^{\alpha}\mathcal{J}^{\beta}\mathcal{H}(t) = \mathcal{J}^{\alpha+\beta}\mathcal{H}(t),\\
&\mathcal{J}^{\alpha}t^{\gamma} = \tfrac{\Gamma(\gamma+1)}{\Gamma(\gamma+\alpha+1)}t^{\gamma+\alpha}, \gamma > -1,\\
&\partial^{\alpha}\mathcal{J}^{\alpha}\mathcal{H}(t) = \mathcal{H}(t),\\
&\mathcal{J}^{\alpha}\partial^{\alpha}\mathcal{H}(t) = \mathcal{H}(t) - \sum_{j=0}^{z-1}\mathcal{H}^{(j)}(0)\tfrac{t^j}{j!},\\
&\partial^{\alpha}t^{\gamma} = 0, \quad \gamma = 0, 1, ..., z-1,\\
&\partial^{\alpha}t^{\gamma} = \tfrac{\Gamma(\gamma+1)}{\Gamma(\gamma+1-z)}t^{\gamma-\alpha}, \quad \gamma \in N \ \ and \ \ \gamma \ge z \ \ or \ \ \gamma \notin N \ \ and \ \ \gamma > z-1.
\end{aligned} \tag{4}$$

Bernoulli polynomials have applications in mathematical analysis, number theory, and other fields of mathematics. Bernoulli polynomials have recently attracted the attention of researchers for their applications in the field of numerical methods. In the following, we introduce Bernoulli polynomials and some of their properties. We also refer to Lehmer (1988); Napoli (2016) for more details and proofs.

Definition 2.2 We can obtain the Bernoulli polynomials $\mathfrak{B}_n(t), n = 0, 1, 2, ...$ using the following expansion Lehmer (1988):

$$\sum_{k=0}^{n}\binom{n+1}{k}\mathfrak{B}_k(t) = (n+1)t^n. \tag{5}$$

Lemma 2.3 *We can obtain the Riemann–Liouville fractional integral of the Bernoulli polynomials $\mathfrak{B}_n(t)$ of arbitrary order $\alpha > 0$ by using the following expansion:*

$$\sum_{k=0}^{n}\binom{n+1}{k}\mathcal{J}^{\alpha}\mathfrak{B}_k(t) = \frac{(n+1)!}{\Gamma(n+\alpha+1)}t^{n+\alpha}, \qquad n = 0, 1, 2, ... \ . \tag{6}$$

Proof Using (4) and (5), the proof is straightforward. □

Bernoulli polynomials and applications of the Riemann–Liouville fractional integral on them can be easily obtained with the help of (5) and (6).

Another way to obtain Bernoulli polynomials is to use their explicit form which is given in Lehmer (1988) as follows:

$$\mathfrak{B}_n(t) = \Sigma_{k=0}^{n}\binom{n}{k}B_{n-k}t^k, \tag{7}$$

where B_n are the Bernoulli numbers. The explicit form of the Bernoulli numbers are given in Gould (1972) as

$$B_n = \Sigma_{k=0}^{n}\frac{1}{k+1}\sum_{j=0}^{k}(-1)^j\binom{k}{j}j^n, \quad n = 1, 2, ..., \tag{8}$$

where $B_0 = 1$. Hence, by utilizing (4) and (7), it is possible to compute the explicit expression for the Riemann–Liouville fractional integral of the Bernoulli polynomials $\mathfrak{B}_n(t)$ of any order $\alpha > 0$ in the following manner:

$$\mathcal{J}^{\alpha}\mathfrak{B}_n(t) = \sum_{k=0}^{n} \binom{n}{k} B_{n-k} \frac{\Gamma(k+1)}{\Gamma(k+\alpha+1)} t^{k+\alpha}. \tag{9}$$

3 Numerical method

In this section, we present the numerical method of solving equation (1) using Bernoulli polynomials.

In the first step, the following successive iterative scheme is used to deal with the nonlinearity of the problem (1):

$$\begin{cases} \partial^{\alpha} u^{n}(t) = \mathcal{G}(t, u^{n-1}(t), \partial^{\beta_1} u^{n-1}(t), \partial^{\beta_2} u^{n-1}(t), ..., \partial^{\beta_m} u^{n-1}(t)), & 0 \le t \le 1, \\ u^{n(i)}(0) = u_i, \quad i = 0, 1, 2, ..., z-1. \end{cases} \tag{10}$$

The convergence of this iterative scheme will be investigated in the next section. Now, let α be the highest order of the derivative of the unknown function in the problem and $\lceil \alpha \rceil$ be the smallest integer greater than or equal to α. We consider the approximation of the derivative of the unknown function of order $\lceil \alpha \rceil$ using Bernoulli polynomials as follows

$$\partial^{\lceil \alpha \rceil} u_N^n(t) = \sum_{j=0}^{N} c_j^n \mathfrak{B}_j(t). \tag{11}$$

Then, applying the operator $\mathcal{J}^{\lceil \alpha \rceil}$ to both sides of (11) and using (4) and the initial conditions given in (10) we have

$$u_N^n(t) = \sum_{j=0}^{N} c_j^n \mathcal{J}^{\lceil \alpha \rceil} \mathfrak{B}_j(t) + \sum_{k=0}^{\lceil \alpha \rceil - 1} u_k \frac{t^k}{k!}. \tag{12}$$

Then we have

$$\partial^{\alpha} u_N^n(t) = \sum_{j=0}^{N} c_j^n \mathcal{J}^{\lceil \alpha \rceil - \alpha} \mathfrak{B}_j(t), \tag{13}$$

and for any $\beta_i, i = 1, ..., m$ we have

$$\partial^{\beta_i} u_N^n(t) = \sum_{j=0}^{N} c_j^n \mathcal{J}^{\lceil \alpha \rceil - \beta_i} \mathfrak{B}_j(t) + \sum_{k=\lceil \beta_i \rceil}^{\lceil \alpha \rceil - 1} u_k \frac{t^{k-\beta_i}}{\Gamma(-\beta_i + k + 1)}. \tag{14}$$

Now substitution (12), (13) and (14) in (10) gives

$$\begin{aligned}\sum_{j=0}^{N} c_j^n \mathcal{J}^{\lceil\alpha\rceil-\alpha}\mathfrak{B}_j(t) \simeq \mathcal{G}\Big(t,\ & \sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil}\mathfrak{B}_j(t) + \sum_{k=0}^{\lceil\alpha\rceil-1} u_k \tfrac{t^k}{k!},\\ & \sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil-\beta_1}\mathfrak{B}_j(t) + \sum_{k=\lceil\beta_1\rceil}^{\lceil\alpha\rceil-1} u_k \tfrac{t^{k-\beta_1}}{\Gamma(-\beta_1+k+1)},\\ & \sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil-\beta_2}\mathfrak{B}_j(t) + \sum_{k=\lceil\beta_2\rceil}^{\lceil\alpha\rceil-1} u_k \tfrac{t^{k-\beta_2}}{\Gamma(-\beta_2+k+1)}, \dots,\\ & \sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil-\beta_m}\mathfrak{B}_j(t) + \sum_{k=\lceil\beta_m\rceil}^{\lceil\alpha\rceil-1} u_k \tfrac{t^{k-\beta_m}}{\Gamma(-\beta_m+k+1)}\Big).\end{aligned} \tag{15}$$

Here, we consider the following shifted Chebyshev points on [0, 1] as collocation points:

$$t_i = \frac{1+cos(i\pi/N)}{2},\ i = 0, 1, ..., N. \tag{16}$$

We want to determine the unknown coefficients such that in (15), the equality is satisfied at the collocation points. Therefore, in each iteration, we determine the unknown coefficients $\boldsymbol{c}^n = (c_0^n, c_1^n, ..., c_N^n)^T$ using the following collocation scheme:

$$\begin{aligned}\sum_{j=0}^{N} c_j^n \mathcal{J}^{\lceil\alpha\rceil-\alpha}\mathfrak{B}_j(t_i) = \mathcal{G}\Big(t_i,\ & \sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil}\mathfrak{B}_j(t_i) + \sum_{k=0}^{\lceil\alpha\rceil-1} u_k \tfrac{t_i^k}{k!},\\ & \sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil-\beta_1}\mathfrak{B}_j(t_i) + \sum_{k=\lceil\beta_1\rceil}^{\lceil\alpha\rceil-1} u_k \tfrac{t_i^{k-\beta_1}}{\Gamma(-\beta_1+k+1)},\\ & \sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil-\beta_2}\mathfrak{B}_j(t_i) + \sum_{k=\lceil\beta_2\rceil}^{\lceil\alpha\rceil-1} u_k \tfrac{t_i^{k-\beta_2}}{\Gamma(-\beta_2+k+1)}, \dots,\\ & \sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil-\beta_m}\mathfrak{B}_j(t_i) + \sum_{k=\lceil\beta_m\rceil}^{\lceil\alpha\rceil-1} u_k \tfrac{t_i^{k-\beta_m}}{\Gamma(-\beta_m+k+1)}\Big),\end{aligned} \tag{17}$$

for $i = 0, 1, ..., N$.

Let $\boldsymbol{c}^0 = (c_0^0, c_1^0, ..., c_N^0)^T$ be an initial guess. At last, to find the values of the unknown coefficients $\boldsymbol{c}^n = (c_0^n, c_1^n, ..., c_N^n)^T$, we must solve a system of linear equations in each iteration:

$$\mathcal{B}\boldsymbol{c}^n = \mathfrak{G}^{n-1}, \tag{18}$$

where the matrix $\mathcal{B}$ has the entries $b_{i,j} = \mathcal{J}^{\lceil\alpha\rceil-\alpha}\mathfrak{B}_j(t_i)$ and $\mathfrak{G}^{n-1} = \left(G_0^{n-1}, G_1^{n-1}, ..., G_N^{n-1}\right)^T$ where

$$\begin{aligned} G_i^{n-1} = \mathcal{G}\Bigg(t_i,\ &\sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil}\mathfrak{B}_j(t_i) + \sum_{k=0}^{\lceil\alpha\rceil-1} u_k\frac{t_i^k}{k!},\\ &\sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil-\beta_1}\mathfrak{B}_j(t_i) + \sum_{k=\lceil\beta_1\rceil}^{\lceil\alpha\rceil-1} u_k\frac{t_i^{k-\beta_1}}{\Gamma(-\beta_1+k+1)},\\ &\sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil-\beta_2}\mathfrak{B}_j(t_i) + \sum_{k=\lceil\beta_2\rceil}^{\lceil\alpha\rceil-1} u_k\frac{t_i^{k-\beta_2}}{\Gamma(-\beta_2+k+1)}, ...,\\ &\sum_{j=0}^{N} c_j^{n-1}\mathcal{J}^{\lceil\alpha\rceil-\beta_m}\mathfrak{B}_j(t_i) + \sum_{k=\lceil\beta_m\rceil}^{\lceil\alpha\rceil-1} u_k\frac{t_i^{k-\beta_m}}{\Gamma(-\beta_m+k+1)}\Bigg). \end{aligned} \tag{19}$$

4 Convergence analysis

This section will thoroughly analyze the convergence of the proposed method under specified conditions. According to Chapter 3 of Gil et al. (2007), we have the following lemma.

Lemma 4.1 *Let $\mathcal{W}_N$ be the polynomial interpolant of the function $\mathcal{W} \in C^{N+1}[0, 1]$ corresponds to the $N + 1$ shifted Chebyshev points (16). Then we have:*

$$\|\mathcal{W} - \mathcal{W}_N\|_\infty \leq \frac{1}{2^{2N+1}(N+1)!}\|\mathcal{W}\|_\infty. \tag{20}$$

Lemma 4.2 *For $\alpha > \beta \geq 0$, $t > 0$ and a function $\mathcal{H}$ with $\mathcal{H}^{(i)}(0) = 0$, $i = 0, 1, 2, ..., \lceil\alpha\rceil-1$ we have*

$$|\partial^\beta\mathcal{H}(t)| \leq \mathcal{J}^{\alpha-\beta}|\partial^\alpha\mathcal{H}(t)|. \tag{21}$$

Proof From (4) and the given conditions, we have

$$\partial^\beta\mathcal{H}(t) = \partial^\beta(\mathcal{J}^\alpha\partial^\alpha\mathcal{H}(t)) = \partial^\beta\mathcal{J}^\beta\mathcal{J}^{\alpha-\beta}\partial^\alpha\mathcal{H}(t) = \mathcal{J}^{\alpha-\beta}\partial^\alpha\mathcal{H}(t). \tag{22}$$

The result is obtained by taking the absolute value of both sides. □

Theorem 4.3 *Assume that the function $\mathcal{G}$ fulfills the following generalized Lipschitz condition:*

$$\begin{aligned} |\mathcal{G}(t, u_0, u_1, ..., u_m) - \mathcal{G}(t, v_0, v_1, ..., v_m)| &\leq L_0|u_0 - v_0| + L_1|u_1 - v_1| + ...\\ &+L_m|u_m - v_m|, \end{aligned} \tag{23}$$

with $L_j \geq 0$, $j = 0, 1, ..., m$. Let u be the exact solution of (1). Furthermore, assume that u_N^{n+1} be the approximated solution achieved through the suggested approach using the starting guess $\mathbf{c}^0 = 0$. Then u_N^{n+1} converges to u when n and N tend to infinity.

Proof For any $t \in [0,1]$ from (1) we have

$$
\begin{aligned}
|\partial^\alpha u(t) - \partial^\alpha u_N^n(t)| &= |\mathcal{G}(t, u(t), \partial^{\beta_1} u(t), ..., \partial^{\beta_m} u(t)) - \partial^\alpha u_N^n(t)| \\
&= |\mathcal{G}(t, u(t), \partial^{\beta_1} u(t), ..., \partial^{\beta_m} u(t)) \\
&\quad -\mathcal{G}(t, u_N^{n-1}(t), \partial^{\beta_1} u_N^{n-1}(t), ..., \partial^{\beta_m} u_N^{n-1}(t))| \\
&\quad +\mathcal{G}(t, u_N^{n-1}(t), \partial^{\beta_1} u_N^{n-1}(t), ..., \partial^{\beta_m} u_N^{n-1}(t)) - \partial^\alpha u_N^n(t)| \\
&= |\mathcal{G}(t, u(t), \partial^{\beta_1} u(t), ..., \partial^{\beta_m} u(t)) \\
&\quad -\mathcal{G}(t, u_N^{n-1}(t), \partial^{\beta_1} u_N^{n-1}(t), ..., \partial^{\beta_m} u_N^{n-1}(t))| \\
&\quad +|\partial^\alpha u_N^n(t) - \mathcal{G}(t, u_N^{n-1}(t), \partial^{\beta_1} u_N^{n-1}(t), ..., \partial^{\beta_m} u_N^{n-1}(t))|.
\end{aligned}
$$

Let $h_n(t) = \mathcal{G}(t, u_N^{n-1}(t), \partial^{\beta_1} u_N^{n-1}(t), \partial^{\beta_2} u_N^{n-1}(t), ..., \partial^{\beta_m} u_N^{n-1}(t))$. According to (13) and (17), the last term on the right-hand side of the above inequality is the interpolation error of the function h_n at the collocation points (16), which will be denoted by $E_{n,N}(t)$ in the following. Then, from Lemma 4.2 and the generalized Lipschitz condition we have

$$
\begin{aligned}
|\partial^\alpha u(t) - \partial^\alpha u_N^n(t)| &\le L_0 |u(t) - u_N^{n-1}(t)| + L_1 |\partial^{\beta_1}(u(t) - u_N^{n-1}(t))| + \\
&\quad ... + L_m |\partial^{\beta_m}(u(t) - u_N^{n-1}(t))| + E_{n,N}(t) \\
&= \sum_{j=0}^{m} L_j |\partial^{\beta_j}(u - u_N^{n-1})| + E_{n,N}(t) \\
&\le \sum_{j=0}^{m} L_j \mathcal{J}^{\alpha-\beta_j} |\partial^\alpha (u - u_N^{n-1})| + E_{n,N}(t) \\
&\le \sum_{j_1=0}^{m} L_{j_1} \mathcal{J}^{\alpha-\beta_{j_1}} \left(\sum_{j_2=0}^{m} L_{j_2} \mathcal{J}^{\alpha-\beta_{j_2}} |\partial^\alpha (u - u_N^{n-2})| + E_{n-1,N}(t) \right) \\
&\quad + E_{n,N}(t) \\
&\le \mathcal{L}^2 \sum_{j_1=0}^{m} \sum_{j_2=0}^{m} \mathcal{J}^{2\alpha - \sum_{k=1}^{2} \beta_{j_k}} |\partial^\alpha (u - u_N^{n-2})| \\
&\quad + E_{n,N}(t) + \mathcal{L} \sum_{j_1=0}^{m} \mathcal{J}^{\alpha-\beta_{j_1}} E_{n-1,N}(t) \\
&\le \mathcal{L}^3 \sum_{j_1=0}^{m} \sum_{j_2=0}^{m} \sum_{j_3=0}^{m} \mathcal{J}^{3\alpha - \sum_{k=1}^{3} \beta_{j_k}} |\partial^\alpha (u - u_N^{n-3})| \\
&\quad + E_{n,N}(t) + \mathcal{L} \sum_{j_1=0}^{m} \mathcal{J}^{\alpha-\beta_{j_1}} E_{n-1,N}(t) \\
&\quad + \mathcal{L}^2 \sum_{j_1=0}^{m} \sum_{j_2=0}^{m} \mathcal{J}^{2\alpha - \sum_{k=1}^{2} \beta_{j_k}} E_{n-2,N}(t) \\
&\vdots \\
&\le \mathcal{L}^n \sum_{j_1=0}^{m} \sum_{j_2=0}^{m} ... \sum_{j_n=0}^{m} \mathcal{J}^{n\alpha - \sum_{k=1}^{n} \beta_{j_k}} |\partial^\alpha (u - u_N^{0})|
\end{aligned}
$$

$$
\begin{aligned}
&+E_{n,N}(t) + \mathcal{L} \sum_{j_1=0}^{m} \mathcal{J}^{\alpha-\beta_{j_1}} E_{n-1,N}(t) \\
&+\mathcal{L}^2 \sum_{j_1=0}^{m} \sum_{j_2=0}^{m} \mathcal{J}^{2\alpha - \sum_{k=1}^{2} \beta_{j_k}} E_{n-2,N}(t) + \ldots \\
&+\mathcal{L}^{n-1} \sum_{j_1=0}^{m} \sum_{j_2=0}^{m} \ldots \sum_{j_{n-1}=0}^{m} \mathcal{J}^{(n-1)\alpha - \sum_{k=1}^{n-1} \beta_{j_k}} E_{1,N}(t) \\
\leq\, & \mathcal{L}^n \|\partial^\alpha u\|_\infty \sum_{j_1=0}^{m} \sum_{j_2=0}^{m} \ldots \sum_{j_n=0}^{m} \frac{t^{\left(n\alpha - \sum_{k=1}^{n} \beta_{j_k}\right)}}{\Gamma\left(n\alpha - \sum_{k=1}^{n} \beta_{j_k} + 1\right)} \\
&+E_N\Bigg(1 + \mathcal{L} \sum_{j_1=0}^{m} \frac{t^{\alpha-\beta_{j_1}}}{\Gamma(\alpha - \beta_{j_1} + 1)} \\
&+\mathcal{L}^2 \sum_{j_1=0}^{m} \sum_{j_2=0}^{m} \frac{t^{2\alpha - \sum_{k=1}^{2} \beta_{j_k}}}{\Gamma(2\alpha - \sum_{k=1}^{2} \beta_{j_k} + 1)} + \ldots \\
&+\mathcal{L}^{n-1} \sum_{j_1=0}^{m} \sum_{j_2=0}^{m} \ldots \sum_{j_{n-1}=0}^{m} \frac{t^{(n-1)\alpha - \sum_{k=1}^{n-1} \beta_{j_k}}}{\Gamma((n-1)\alpha - \sum_{k=1}^{n-1} \beta_{j_k} + 1)}\Bigg) \\
\leq\, & (\mathcal{L}(m+1))^n \|\partial^\alpha u\|_\infty \frac{1}{\Gamma\left(n(\alpha-\beta)+1\right)} \\
&+E_N\Bigg(1 + \mathcal{L}(m+1) \frac{1}{\Gamma(\alpha - \beta + 1)} \\
&+(\mathcal{L}(m+1))^2 \frac{1}{\Gamma(2(\alpha-\beta)+1)} + \ldots \\
&+(\mathcal{L}(m+1))^{n-1} \frac{1}{\Gamma((n-1)(\alpha-\beta)+1)}\Bigg) \\
\leq\, & (\mathcal{L}(m+1))^n \|\partial^\alpha u\|_\infty \frac{1}{\Gamma\left(n(\alpha-\beta)+1\right)} \\
&+E_N \sum_{k=0}^{\infty} \frac{(\mathcal{L}(m+1))^k}{\Gamma(k(\alpha-\beta)+1)} \\
=\, & \frac{(\mathcal{L}(m+1))^n}{\Gamma\left(n(\alpha-\beta)+1\right)} \|\partial^\alpha u\|_\infty + E_N \mathbf{E}_{\alpha-\beta}(\mathcal{L}(m+1)),
\end{aligned}
$$

where $\beta_0 = 0, \beta = \max_{0\leq j\leq m} \beta_j$, $E_N = \max_{1\leq j\leq n} \|E_{j,N}\|_\infty$, $\mathcal{L} = \max_{0\leq j\leq m} L_j$ and $\mathbf{E}_{\alpha-\beta}(t)$ is the one parameter Mittag-Leffler function. From Lemma 4.1 and Lemma 4.2 we can see that

$$\begin{aligned}|u(t)-u_N^n(t)| &\le \mathcal{J}^\alpha|\partial^\alpha(u(t)-u_N^n(t))|\\ &\le \mathcal{J}^\alpha\left(\frac{(\mathcal{L}(m+1))^n}{\Gamma(n(\alpha-\beta)+1)}\|\partial^\alpha u\|_\infty+\frac{1}{2^{2N+1}(N+1)!}\mathcal{C}\right)\\ &\le \frac{1}{\Gamma(\alpha+1)}\left(\frac{(\mathcal{L}(m+1))^n}{\Gamma(n(\alpha-\beta)+1)}\|\partial^\alpha u\|_\infty+\frac{1}{2^{2N+1}(N+1)!}\mathcal{C}\right),\end{aligned}$$

which shows the convergence of the proposed method as $n\to\infty$, $N\to\infty$. □

5 Numerical results

In this section, to check the efficiency and capability of the method, several examples have been solved using the proposed method, and the obtained results are presented. For this purpose, the results obtained from the implementation of the method have been compared with the exact solutions, and the results obtained by other methods reported in the articles. Also, the execution times of the method are reported in the tables. In this section, we have utilized an N-by-1 vector of zeros, denoted as $\boldsymbol{c}^0=(0,0,...,0)^T$, as the initial guess for all the examples. The results were obtained using MATLAB 2015a on a computer with the following specifications: Processor: 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz; memory (RAM): 8.00 GB; and system type: 64-bit operating system, x64-based processor.

Example 5.1 Consider the following nonlinear problem containing multi-fractional-order derivatives as the first example Izadi and Cattani (2020); Bhrawy et al. (2015):

$$\begin{aligned}\partial^\alpha u(t)\ &+\partial^{\beta_1}u(t)\partial^{\beta_2}u(t)+(u(t))^2=\\ &t^6+\frac{6t^{3-alpha}}{\Gamma(4-\alpha)}+\frac{36t^{6-\beta_1-\beta_2}}{\Gamma(4-\beta_1)\Gamma(4-\beta_2)},\qquad t\in[0,1],\end{aligned}\tag{24}$$

along with the following initial conditions

$$u(0)=\frac{du}{dt}|_{t=0}=\frac{d^2u}{dt^2}|_{t=0}=0,\tag{25}$$

where $2<\alpha<3$, $0<\beta_1<1$ and $1<\beta_2<2$. Also, $u(t)=t^3$ is the exact solution to the above problem. We consider this example in the following five cases:

Case 1: $\alpha=5/2$, $\beta_1=9/10$, $\beta_2=3/2$.
Case 2: $\alpha=2.000001$, $\beta_1=0.000009$, $\beta_2=1.000001$.
Case 3: $\alpha=2.99$, $\beta_1=0.99$, $\beta_2=1.99$.
Case 4: $\alpha=2.75$, $\beta_1=0.75$, $\beta_2=1.75$.
Case 5: $\alpha=2.9999$, $\beta_1=0.9999$, $\beta_2=1.9999$.

The maximum absolute errors of the approximate solutions of example 5.1 obtained by the proposed method are given in Table 1. Also in this table, some comparisons with the results presented in Izadi and Cattani (2020); Bhrawy et al. (2015) are given. The Fig. 1 shows the maximum absolute errors versus n with $N=20$ and versus N with $n=20$ for case 1 in logarithmic scale. As can be seen, the accuracy of the approximate solutions increases with increasing N and n.

Example 5.2 As the second example, consider the following nonlinear, multi-fractional-order equation with variable coefficients:

$$\begin{aligned}\partial^2u(t)\ &+\Gamma(\tfrac{4}{5})\sqrt[5]{t^6}\partial^{\frac{6}{5}}u(t)+\tfrac{11}{9}\sqrt[6]{t}\partial^{\frac{1}{6}}u(t)=\\ &(\partial^1u(t))^2+2+\tfrac{1}{10}t^2,\qquad 0\le t\le 1,\end{aligned}\tag{26}$$

Table 1 The maximum absolute errors using $N = 20$ and $n = 20$ (Example 5.1)

	Case 1	Case 2	Case 3	Case 4	Case 5
Our Method	3.3307×10^{-16}	2.2204×10^{-16}	2.2205×10^{-16}	4.4409×10^{-16}	3.3307×10^{-16}
Ref. Izadi and Cattani (2020)	6.8508×10^{-14}	3.8221×10^{-15}	1.8762×10^{-16}	3.4801×10^{-14}	4.4918×10^{-16}
Ref. Bhrawy et al. (2015)	3.15×10^{-5}	6.29×10^{-12}	1.95×10^{-5}	1.06×10^{-4}	2.06×10^{-7}
CPU time(s)	1.24	1.22	1.2	1.32	1.23

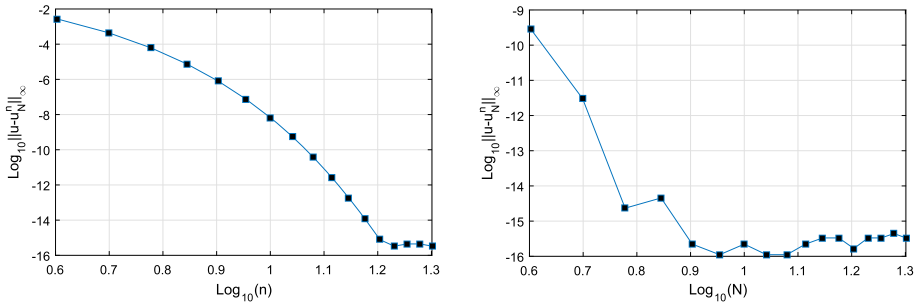

Fig. 1 The maximum absolute errors versus n with $N = 20$(left) and versus N with $n = 20$(right) for Example 5.1 case 1

along with the following conditions

$$u(0) = \frac{du}{dt}|_{t=0} = 0. \tag{27}$$

$u(t) = 1 + t^2$ is the exact solution to the above problem. The maximum absolute errors of the approximate solutions of example 5.2 obtained by the proposed method are given in Table 2. As can be seen, the accuracy of the approximate solutions increases with increasing N and n.

Example 5.3 The following nonlinear, multi-order, non-homogenous fractional differential equation is considered in the third example Shiralashetti and Deshi (2016); El-Sayed et al. (2010); Khan et al. (2022):

$$\begin{aligned} a\partial^{\alpha}u &+ b\partial^{\beta_2}u + c\partial^{\beta_1}u + eu^3 = \\ &\tfrac{2a}{\Gamma(4-\alpha)}t^{3-\alpha} + \tfrac{2b}{\Gamma(4-\beta_2)}t^{3-\beta_2} + \tfrac{2c}{\Gamma(4-\beta_1)}t^{3-\beta_1} + \tfrac{e}{27}t^9, \qquad 0 \le t \le 1, \end{aligned} \tag{28}$$

along with the conditions

$$\frac{d^i u}{dt^i}|_{t=0} = 0, \qquad i = 0, 1, ..., \lceil\alpha\rceil - 1, \tag{29}$$

where $0 < \beta_1 < 1$ and $0 < \beta_1 < \beta_2 < \lceil\alpha\rceil - 1$. Also, $u(t) = \frac{t^3}{3}$ is the exact solution to the above problem. We consider this example in the following cases:

Case 1: Shiralashetti and Deshi (2016); El-Sayed et al. (2010) $a = e = 1, b = 2, c = 0.5, \alpha = 2, \beta_1 = 0.00196, \beta_2 = 0.07621$.

Case 2: Shiralashetti and Deshi (2016); El-Sayed et al. (2010) $a = 1, b = 0.1, c = 0.2, e = 0.3, \alpha = 2, \beta_1 = \sqrt{5}/5, \beta_2 = \sqrt{2}/2$.

Case 3: Shiralashetti and Deshi (2016); Khan et al. (2022) $a = b = c = e = 1, \alpha = 2.2, \beta_1 = 0.75, \beta_2 = 1.25$.

The maximum absolute errors of the approximate solutions of example 5.3 obtained by the proposed method are given in Table 3. Also in this table, some comparisons with the results presented in Shiralashetti and Deshi (2016); El-Sayed et al. (2010); Khan et al. (2022) are given.

Example 5.4 Consider the following problem with variable coefficients Talib et al. (2022):

$$\partial^2 u(t) + \sqrt{t^7}\partial^{\frac{3}{2}}u(t) + (u(t))^2 = 2 + t^4 + \frac{4t^4}{\sqrt{\pi}}, \qquad 0 \le t \le 1, \tag{30}$$

Table 2 The maximum absolute errors using different values of N and n (Example 5.2)

N, n	$n = 5$	$n = 10$	$n = 15$	$n = 20$
$N = 5$	2.5245×10^{-5}	5.0835×10^{-8}	7.8558×10^{-11}	1.0392×10^{-13}
CPU time(s)	0.97	0.985	0.99	1.001
$N = 10$	1.3666×10^{-5}	1.2511×10^{-10}	6.6613×10^{-16}	4.4409×10^{-16}
CPU time(s)	1.02	1.04	1.05	1.08
$N = 15$	1.3666×10^{-5}	1.2237×10^{-10}	4.4409×10^{-16}	4.4409×10^{-16}
CPU time(s)	1.13	1.17	1.16	1.13
$N = 20$	1.3666×10^{-5}	1.2237×10^{-10}	4.4409×10^{-16}	4.4409×10^{-16}
CPU time(s)	1.15	1.18	1.2	1.24

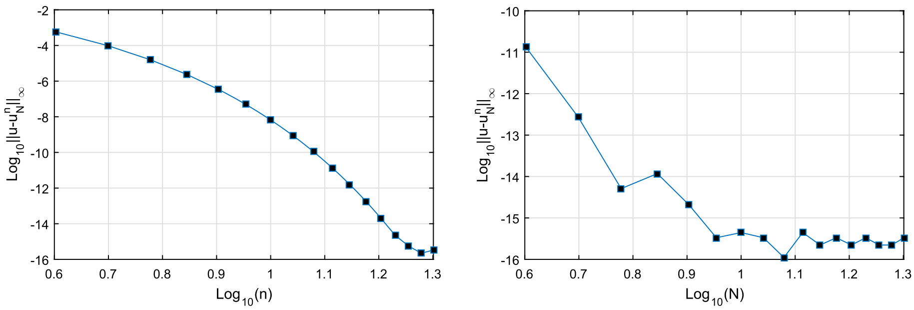

Fig. 2 The maximum absolute errors versus n with $N = 20$(left) and versus N with $n = 20$(right) for Example 5.4

along with the conditions

$$u(0) = \frac{du}{dt}|_{t=0} = 0. \tag{31}$$

$u(t) = t^2$ is the exact solution to the above problem. The maximum absolute errors of the approximate solutions of example 5.4 obtained by the proposed method using different values of N and n are given in Table 4. The Fig. 2 shows the maximum absolute errors versus n with $N = 20$ and versus N with $n = 20$ in logarithmic scale. The best-reported results in Talib et al. (2022) have more than 10^{-10} absolute error.

Example 5.5 Consider the following nonlinear multi-order fractional differential equation:

$$\partial^{\alpha} u + \partial^{\beta_2} u + \partial^{\beta_1} u + u^3 = f(t), \qquad 0 \leq t \leq 1, \tag{32}$$

along with the conditions

$$\frac{d^i u}{dt^i}|_{t=0} = 0, \qquad i = 0, 1, ..., \lceil \alpha \rceil - 1, \tag{33}$$

where $0 < \beta_1 < 1$ and $0 < \beta_1 < \beta_2 < \lceil \alpha \rceil - 1$. The function f can be obtained corresponding to the exact solution $u(t) = t^3 cos(t)$ and the following cases:

Case 1: $\alpha = 2.2$, $\beta_1 = 0.75$, $\beta_2 = 1.25$.

Case 2: $\alpha = 2$, $\beta_1 = \sqrt{5}/5$, $\beta_2 = \sqrt{2}/2$.

Table 3 The maximum absolute errors using $N = 20$ and $n = 10$.(Example 5.3)

	Our Method	Ref. Shiralashetti and Deshi (2016)	Ref. El-Sayed et al. (2010)	Ref. Khan et al. (2022)	Ref. Zhang et al. (2021)	Ref. Chen et al. (2020)
Case 1	1.6653×10^{-16}	1.5457×10^{-6}	3.99235×10^{-4}	–	8.156×10^{-3}	2.7649×10^{-14}
CPU time(s)	1.2					
Case 2	4.4409×10^{-16}	2.2990×10^{-6}	3.88881×10^{-4}	–	–	–
CPU time(s)	1.13					
Case 3	1.1102×10^{-16}	2.3923×10^{-6}	–	2.10×10^{-6}	–	–
CPU time(s)	1.45					

Table 4 The maximum absolute errors using different values of N and n.(Example 5.4)

N, n	$n=5$	$n=10$	$n=15$	$n=20$
$N=5$	1.3096×10^{-4}	3.6017×10^{-7}	4.6356×10^{-10}	2.6956×10^{-13}
CPU time(s)	0.98	1.04	1.05	1.07
$N=10$	9.7718×10^{-5}	6.7023×10^{-9}	6.0485×10^{-13}	4.4409×10^{-16}
CPU time(s)	1.04	1.05	1.06	1.08
$N=15$	9.7636×10^{-5}	6.6369×10^{-9}	1.9496×10^{-13}	3.3307×10^{-16}
CPU time(s)	1.05	1.06	1.08	1.10
$N=20$	9.7636×10^{-5}	6.6309×10^{-9}	1.8274×10^{-13}	3.3307×10^{-16}
CPU time(s)	1.07	1.08	1.10	1.12

Table 5 The maximum absolute errors using different values of N and n.(Example 5.5, Case 1)

N, n	$n=5$	$n=10$	$n=15$
$N=5$	2.1504×10^{-6}	2.1504×10^{-6}	2.1504×10^{-6}
CPU time(s)	0.96	1.02	1.2
$N=10$	4.2910×10^{-14}	4.2688×10^{-14}	4.2910×10^{-14}
CPU time(s)	1.0	1.13	1.23
$N=15$	5.5511×10^{-16}	4.9960×10^{-16}	4.4409×10^{-16}
CPU time(s)	1.17	1.28	1.31
$N=20$	7.7716×10^{-16}	4.4409×10^{-16}	4.4409×10^{-16}
CPU time(s)	1.21	1.32	1.41

Table 6 The maximum absolute errors using different values of N and n.(Example 5.5, Case 2)

N, n	$n=5$	$n=10$	$n=15$
$N=5$	3.2172×10^{-5}	3.2172×10^{-5}	3.2172×10^{-5}
CPU time(s)	0.95	1.10	1.23
$N=10$	1.1080×10^{-12}	1.1078×10^{-12}	1.1078×10^{-12}
CPU time(s)	1.07	1.17	1.28
$N=15$	2.1761×10^{-14}	2.2204×10^{-16}	2.2204×10^{-16}
CPU time(s)	1.20	1.23	1.33
$N=20$	2.1538×10^{-14}	4.4409×10^{-16}	2.2204×10^{-16}
CPU time(s)	1.22	1.29	1.46

The maximum absolute errors of the approximate solutions of example 5.5 obtained by the proposed method using different values of N and n are given in Tables 5 and 6.

Example 5.6 Finally, consider the following nonlinear multi-order fractional differential equation:

$$\partial^{\alpha} u+\partial^{\beta} u+e^{u}=f(t), \qquad 0 \leq t \leq 1, \tag{34}$$

Table 7 The maximum absolute errors using different values of N and n.(Example 5.6, Case 1)

N, n	$n = 5$	$n = 10$	$n = 15$
$N = 5$	2.1535×10^{-3}	2.1534×10^{-3}	2.1534×10^{-3}
CPU time(s)	1.21	1.33	1.48
$N = 10$	5.7443×10^{-4}	5.7443×10^{-4}	5.7443×10^{-4}
CPU time(s)	1.47	1.55	1.6
$N = 15$	2.4975×10^{-4}	2.4974×10^{-4}	2.4974×10^{-4}
CPU time(s)	1.56	1.61	1.66
$N = 20$	1.7840×10^{-4}	1.7833×10^{-4}	1.7833×10^{-4}
CPU time(s)	1.63	1.67	1.75

Table 8 The maximum absolute errors using different values of N and n.(Example 5.6, Case 2)

N, n	$n = 5$	$n = 10$	$n = 15$
$N = 5$	9.8749×10^{-4}	9.8748×10^{-4}	9.8748×10^{-4}
CPU time(s)	1.39	1.45	1.58
$N = 10$	2.4885×10^{-4}	2.4882×10^{-4}	2.4882×10^{-4}
CPU time(s)	1.57	1.6	1.68
$N = 15$	1.11559×10^{-4}	1.1158×10^{-4}	1.1155×10^{-4}
CPU time(s)	1.61	1.69	1.74
$N = 20$	5.7533×10^{-5}	5.7442×10^{-5}	5.7438×10^{-5}
CPU time(s)	1.7	1.75	1.83

along with the conditions

$$\frac{d^i u}{dt^i}|_{t=0} = 0, \qquad i = 0, 1, ..., \lceil \alpha \rceil - 1, \tag{35}$$

where $0 < \beta < 1$ and $0 < \beta < \lceil \alpha \rceil - 1$. The function f can be obtained corresponding to the exact solution $u(t) = t^{\alpha}$, which has limited regularity at the beginning of time. Consider the following cases:

Case 1: $\alpha = \sqrt{3}$, $\beta = 0.7$.

Case 2: $\alpha = 1.85$, $\beta = 0.5$.

The maximum absolute errors of the approximate solutions of example 5.6 obtained by the proposed method using different values of N and n are given in Tables 7 and 8. As we can see, we reached acceptable results even with a small number of collocation points and iterations. In the numerical results presented in this section, it can be seen that by increasing the number of collocation points N and iterations n, we can approximate the solutions with higher precision. This point was predicted in the previous given analytical results.

6 Conclusion

In this work, we have presented a practical, efficient, and vigorous method for solving nonlinear multi-fractional-order differential equations with the help of the integrated Bernoulli polynomials. The method is simple in programming and fast in execution. In the presented method, we use the integrated Bernoulli polynomials combined with the simple iteration method and the collocation method to approximate the solution. We have given a convergence analysis of the proposed iterative method for nonlinear problems. Several numerical examples of the implementation of the method are presented to test the method's efficiency, power, and versatility. The presented numerical results and comparisons show the accuracy and efficiency of the proposed method.

Data availability Relevant data can be made available upon reasonable request.

Declarations

Conflict of interest None.

References

Abd-Elhameed WM, Alsuyuti MM (2023) Numerical treatment of multi-term fractional differential equations via new kind of generalized chebyshev polynomials. Fractal Fract 7(1):74
Alipour M, Agahi H (2018) New computational techniques for solving nonlinear problems using g-fractional differential operator. J Comput Appl Math 330:70–74
Alqhtani M, Owolabi KM, Saad KM, Pindza E (2022) Efficient numerical techniques for computing the Riesz fractional-order reaction-diffusion models arising in biology. Chaos, Solitons & Fractals 161:112394
Amin R, Shah K, Mlaiki N, Yüzbaşi Ş, Abdeljawad T, Hussain A (2022) Existence and numerical analysis using Haar wavelet for fourth-order multi-term fractional differential equations. Comput Appl Math 41(7):329
Azarnavid B (2023) The Bernoulli polynomials reproducing kernel method for nonlinear Volterra integro-differential equations of fractional order with convergence analysis. Comput Appl Math 42(1):1–17
Azarnavid B, Emamjomeh M, Nabati M (2022) A shooting like method based on the shifted Chebyshev polynomials for solving nonlinear fractional multi-point boundary value problem. Chaos, Solitons & Fractals 159:112159
Bakhshandeh-Chamazkoti R, Alipour M (2022) Lie symmetries reduction and spectral methods on the fractional two-dimensional heat equation. Math Comput Simul 200:97–107
Bhrawy AH, Tohidi E, Soleymani F (2012) A new Bernoulli matrix method for solving high-order linear and nonlinear Fredholm integro-differential equations with piecewise intervals. Appl Math Comput 219(2):482–497
Bhrawy AH, Taha TM, Machado JAT (2015) A review of operational matrices and spectral techniques for fractional calculus. Nonlinear Dyn 81:1023–1052
Chen X, Lei S, Wang L (2020) Bernoulli Polynomials Collocation Method for Multi-term Fractional Differential Equations. IAENG Int J Appl Math 50(3):1–8
Chen W, Sun H, Li X (2022) Fractional derivative modeling in mechanics and engineering. Springer, New York
Dadkhah E, Shiri B, Ghaffarzadeh H, Baleanu D (2020) Visco-elastic dampers in structural buildings and numerical solution with spline collocation methods. J Appl Math Comput 63:29–57
Diethelm K (2010) The analysis of fractional differential equations: an application-oriented exposition using differential operators of Caputo type, vol 2004. Springer, Berlin
Diethelm K, Ford NJ (2004) Multi-order fractional differential equations and their numerical solutions. Appl Math Comput 154:621–640
Eftekhari T, Rashidinia J (2023) A new operational vector approach for time-fractional subdiffusion equations of distributed order based on hybrid functions. Mathematical Methods in the Applied Sciences 46(1):388–407
El-Sayed AMA, El-Kalla IL, Ziada EAA (2010) Analytical and numerical solutions of multi-term nonlinear fractional orders differential equations. Appl Numer Math 60(8):788–797
Erturk VS, Alomari AK, Kumar P, Murillo-Arcila M (2022) Analytic solution for the strongly nonlinear multi-order fractional version of a BVP occurring in chemical reactor theory, Discrete Dynamics in Nature and Society 2022

Gil A, Segura J, Temme NM (2007) Numerical methods for special functions. Society for Industrial and Applied Mathematics

Gould WH (1972) Explicit formulas for Bernoulli numbers. Am Math Mon 79(1):44–51

Izadi M, Cattani C (2020) Generalized Bessel polynomial for multi-order fractional differential equations. Symmetry 12(8):1260

Joujehi AS, Derakhshan MH, Marasi HR (2022) An efficient hybrid numerical method for multi-term time fractional partial differential equations in fluid mechanics with convergence and error analysis. Commun Nonlinear Sci Numer Simul 114:106620

Khan NA, Ibrahim Khalaf O, Andres Tavera Romero C, Sulaiman M., Bakar M.A. (2022) Application of intelligent paradigm through neural networks for numerical solution of multiorder fractional differential equations, Computational Intelligence and Neuroscience 2022

Khodabandelo HR, Shivanian E, Abbasbandy S (2022) A novel shifted Jacobi operational matrix method for nonlinear multi-terms delay differential equations of fractional variable-order with periodic and anti-periodic conditions. Mathematical Methods in the Applied Sciences 45(16):10116–10135

Lehmer DH (1988) A new approach to Bernoulli polynomials. Am Math Mon 95(10):905–911

Nagy AM (2022) Numerical solutions for nonlinear multi-term fractional differential equations via Dickson operational matrix. Int J Comput Math 99(7):1505–1515

Napoli A (2016) Solutions of linear second order initial value problems by using Bernoulli polynomials. Appl Numer Math 99:109–120

Odibat Z, Baleanu D (2022) Nonlinear dynamics and chaos in fractional differential equations with a new generalized Caputo fractional derivative. Chin J Phys 77:1003–1014

Postavaru O (2022) Generalized fractional-order hybrid of block-pulse functions and Bernoulli polynomials approach for solving fractional delay differential equations, Soft Computing: 1-13

Postavaru O, Toma A (2022) A numerical approach based on fractional-order hybrid functions of block-pulse and Bernoulli polynomials for numerical solutions of fractional optimal control problems. Math Comput Simul 194:269–284

B. Prakash, A. Setia and S. Bose, Numerical solution for a system of fractional differential equations with applications in fluid dynamics and chemical engineering, International Journal of Chemical Reactor Engineering 15(5) (2017)

Rezabeyk S, Abbasbandy S, Shivanian E, Derili H (2023) A new approach to solve weakly singular fractional-order delay integro-differential equations using operational matrices, Journal of Mathematical Modeling

Shi J, He K, Fang H (2022) Chaos, Hopf bifurcation and control of a fractional-order delay financial system. Math Comput Simul 194:348–364

Shiralashetti SC, Deshi AB (2016) An efficient Haar wavelet collocation method for the numerical solution of multi-term fractional differential equations. Nonlinear Dyn 83:293–303

Talib I, Raza A, Atangana A, Riaz MB (2022) Numerical study of multi-order fractional differential equations with constant and variable coefficients. Journal of Taibah University for Science 16(1):608–620

Verma P, Kumar M (2022) Analytical solution with existence and uniqueness conditions of non-linear initial value multi-order fractional differential equations using Caputo derivative, Engineering with Computers: 1-18

Zhang B, Tang Y, Zhang X (2021) Numerical solution of fractional differential equations using hybrid Bernoulli polynomials and block pulse functions. Mathematical Sciences 15:293–304