Ricerche di Matematica
https://doi.org/10.1007/s11587-021-00650-9

Approximating solutions to fractional-order Bagley-Torvik equation via generalized Bessel polynomial on large domains

Mohammad Izadi[1] · **Şuayip Yüzbaşı**[2] · **Carlo Cattani**[2]

Received: 23 June 2021 / Revised: 11 August 2021 / Accepted: 9 September 2021
© Università degli Studi di Napoli "Federico II" 2021

Abstract
The paper exhibits a practical and effective scheme to approximate the solutions of a class of fractional differential equations known as the Bagley-Torvik equations. The underlying fractional derivative is based on the Caputo definition. Both the boundary and initial conditions are considered while the domain of approximation is taken sufficiently large. The principle idea behind our approximation algorithm is to use a novel class of (orthogonal) polynomials called the Bessel polynomials, which their coefficients are positive. The convergence and error analysis of the Bessel solution are investigated in the L_2 and infinity norms. Representing the unknown solution and its derivatives in terms of basis functions together with collocation points, the Bagley-Torvik equations are reduced into an algebraic form. In particular, we present a simple but fast algorithm with linear complexity for computing the q-th order fractional derivative of the basis functions in the vectorized form. Several practical test problems are given to illustrate the utility and efficiency of the proposed approximation algorithm. Validation of the method is obtained by comparison with existing available numerical solutions. Based on experiments, the present approach produces the results of high accuracy to obtain the approximate solutions of Bagley-Torvik equations especially for long-time computations.

✉ Mohammad Izadi
izadi@uk.ac.ir

Şuayip Yüzbaşı
syuzbasi@akdeniz.edu.tr

Carlo Cattani
cattani@unitus.it

[1] Department of Applied Mathematics, Faculty of Mathematics and Computer, Shahid Bahonar University of Kerman, Kerman, Iran

[2] Department of Mathematics, Faculty of Science, Akdeniz University, TR 07058 Antalya, Turkey

Published online: 07 October 2021

Keywords Caputo fractional derivative · Bessel functions · Collocation method · Error analysis · Fractional-order Bagley-Torvik equation

Mathematics subject classification 65L10 · 26A33 · 65L60 · 42C05 · 65L05

1 Introduction

During the past decades, fractional calculus has been provided more powerful tools in modelling of various realistic phenomena in diverse disciplines of engineering and physical science. This stems from the fact that fractional derivatives and integrals are capable of accounting some memory effects of various materials and processes, see [1,2]. Fractional-order differential equations (FODEs) as outcomes of the most of these models have attracted a great deal of attention by many researchers. The exact analytical solutions of most FODEs cannot be obtained or may be available only in a closed-form. Thus, approximation algorithms or numerical procedures are applied preferably.

In this research, we investigate the Bagley-Torvik equation of fractional order. Historically, the origin of this equation goes backed to Bagley and Torvik who first modelled the motion of a thin plate immersed in a Newtonian fluid while connected to a spring [3], see also [1]. Let assume that m and A denote the mass and surface area of a thin plate immersed in a Newtonian fluid with density ρ and viscosity μ. The plate attached to a spring with stiffness k hanged on a fixed point as shown in Fig. 1. The displacement of the plate denotes by $u(t)$ and satisfies

$$m\,u''(t) + 2A\sqrt{\rho\,\mu}\,D^{(\frac{3}{2})}\,u(t) + k\,u(t) = f(t). \tag{1.1}$$

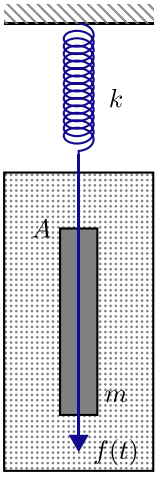

Fig. 1 An immersed plate in a Newtonian fluid

Let us consider the generalized version of the Bagley-Torvik equation (1.1) to an arbitrary $1 < q < 2$ taking the form

$$P_2(t)\, D^2 u(t) + P_1(t)\, D^{(q)} u(t) + P_0(t)\, u(t) = F(t), \quad t \in [0, L], \tag{1.2}$$

with the following initial or boundary conditions

$$u(0) = u_0, \quad u'(0) = u_1, \qquad \text{or} \qquad u(0) = u_0, \quad u(L) = u_1, \tag{1.3}$$

where u_0, u_1 are two given constants, $P_i(t)$ for $i = 0, 1, 2$ and $F(t)$ are some real-valued functions. In (1.2), $D^{(q)}$ stands for the fractional Caputo derivative operator so that $q = \frac{1}{2}$ or $\frac{3}{2}$. Some results about existence as well as the uniqueness of the solution of model problem (1.2) can be found in [1].

Numerous attempts have been undertaken to develop numerical and approximative methods for (1.2) over the past decades. Some recent numerical approximations include, but no limited to, the method based on evolutionary computational intelligence [4], the Gegenbauer wavelet method [5], the Bessel collocation scheme [6], the hybrid of Bernoulli polynomials and block-pulse functions [7], the hybridizable and local discontinuous Galerkin methods [8,9], the sinc, Fermat, and Fibonacci operational matrix methods [10–12], the Müntz-Legendre polynomials [13], the reproducing kernel Hilbert space algorithms [14], and the Galerkin-type approach [15].

Bessel polynomials were defined by H.L. Krall and O. Frink as the (unique) solution of a second-order linear differential equation naturally appears in the study of classical wave equation in spherical coordinates [16]. The Bessel polynomials have many interesting properties and applications, although they are not so popular as the so-called classical polynomials such as Hermit, Jacobi, and Laguerre. The essential characteristic of these polynomials is the positivitness of the coefficients compared to other well-known traditional polynomials. for more informations and applications of these polynomials we refer to the recently published works [17–21].

The process of expanding a given function in terms of orthogonal basis functions is an ubiquitous task in many practical applications in science and engineering. Beside the applications of integer-ordered basis functions, using the fractional-order of these bases has proved to be more efficient especially when the target functions have a fractional-like behaviour, see cf. [22–29]. Generalizing to the fractional-order is usually accomplished by substitution $x = t^\alpha$, where $\alpha > 0$, in the underlying differential equations. In the following, we will give an original numerical method, based on Bessel polynomials, to solve the fractional equation (1.2) on a sufficiently large interval. The main idea is to use these orthogonal functions together with some suitable collocation points in order to convert the differential operator of (1.2) into a system of algebraic form, thus optimizing the computational effort.

The organization of the research is the following. Section 2 deals with the Caputo fractional derivative and some properties of the Caputo fractional operator are reviewed. The Bessel functions and their fractional-order generalization are discussed in Sect. 3. Moreover, their convergence and error analysis are studied in detail with regard to L_2 and infinity norms. In Sect. 4, we describe the methodology of the Bessel-collocation approach to acquire the solution of the underlying model problem.

In Sect. 5, several practical test examples are solved numerically to describe the effectiveness of the presented technique. In Sect. 6, we are shortly reported a brief summary and conclusion.

2 Preliminary remarks

Let us first mention some basic definitions on the Caputo fractional derivative (see also [1,17] and references therein).

Definition 2.1 Let $f(t)$ be an n-times differentiable function. The Caputo fractional derivative $D^{(q)}$ of $f(t)$ of order $q > 0$ is

$$D^{(q)} f(t) = \begin{cases} \mathcal{I}^{(n-q)} f^{(n)}(t) & \text{if } n-1 < q < n, \\ f^{(n)}(t), & \text{if } q = n, \quad n \in \mathbb{N}, \end{cases} \tag{2.1}$$

being

$$\mathcal{I}^{(q)} f(t) = \frac{1}{\Gamma(q)} \int_0^t (t-s)^{q-1} f(s)\, ds, \quad t > 0.$$

This operator owns the following properties

$$D^{(q)}(C) = 0 \quad (C \text{ is a constant}), \tag{2.2}$$

$$D^{(q)} t^p = \begin{cases} \dfrac{\Gamma(p+1)}{\Gamma(p+1-q)} t^{p-q}, & \text{for } p \in \mathbb{N}_0 \text{ and } p \geq \lceil q \rceil, \text{ or } p \notin \mathbb{N}_0 \text{ and } p > \lfloor q \rfloor, \\ 0, & \text{for } p \in \mathbb{N}_0 \text{ and } p < \lceil q \rceil. \end{cases} \tag{2.3}$$

Further properties of this operator can be found in many books like e.g. [1].

3 Generalization of Bessel functions

In this section, we give some preliminary remarks on fractional Bessel polynomials, for more details see e.g. [17] and references therein.

3.1 Integer-order Bessel polynomials

Let us denoted $\mathcal{B}_n(x)$ to be the Bessel polynomial of degree n. It is equal to unity at $x = 0$ and satisfies the linear equation $x^2 y''(x) + (2x+2) y'(x) = n(n+1) y(x)$, where $y(x) := \mathcal{B}_n(x)$. The explicit form of these polynomials as the (unique) solution

of this equation is given by

$$\mathcal{B}_n(x) = \sum_{k=0}^{n} b(n,k)\, x^k, \quad b(n,k) := \binom{n}{k}\binom{n+k}{k}\frac{k!}{2^k}, \quad n = 0, 1, \ldots. \tag{3.1}$$

It can be easily checked that $\mathcal{B}_0(x) = 1$ and $\mathcal{B}_1(x) = 1 + x$. The remaining Bessel polynomials can be also obtained by the recurrence relation

$$\mathcal{B}_{n+1}(x) = (2n+1)x\, \mathcal{B}_n(x) + \mathcal{B}_{n-1}(x), \quad n = 1, 2, \ldots. \tag{3.2}$$

It should be noticed that the coefficients of these polynomials being always positive with $\mathcal{B}_n(0) = 1$ and $\mathcal{B}_n'(0) = n(n+1)/2$. It can be shown that these polynomials are orthogonal with regard to the weighting function $w(x) \equiv e^{-2/x}$ in the sense that

$$\frac{1}{2\pi i}\int_U \mathcal{B}_n(x)\, \mathcal{B}_m(x)\, w(x)dx = \frac{2(-1)^{n+1}\delta_{nm}}{2n+1}, \tag{3.3}$$

where $\delta_{nm} = 1$ if $n = m$ and is 0, otherwise. Here, U is the unite circle. We emphasize that any curve around $x = 0$ can be taken instead of U as the path of integration.

3.2 Non-integer Bessel polynomials

Let us derive the fractional version of the Bessel differential equation by replacing $x = t^\alpha/L, \alpha > 0, L > 0$ in the original equation. Thus, we get

$$t^{1+\alpha}\, y''(t) + \Big[(1+\alpha)t^\alpha + 2\alpha L\Big]\, y'(t) = n(n+1)\,\alpha^2\, t^{\alpha-1}\, y(t). \tag{3.4}$$

Setting $\alpha, L = 1$, we will recover the classical Bessel equation. Let $\mathcal{B}_n^\alpha(t) = \mathcal{B}_n(x)$ denote the polynomial solutions of the modified equation (3.4). By extending them on the interval $[0, L]$ and using (3.1), we will get the following explicit expression

$$\mathcal{B}_n^\alpha(t) = \sum_{k=0}^{n} b_L(n,k)\, t^{k\alpha}, \quad b_L(n,k) := b(n,k)/L^k, \quad 0 \le t \le L < \infty, \tag{3.5}$$

where $b(n,k)$ is defined in (3.1). To prove orthogonality, an easy calculation shows that the differential equation (3.4) can be expressed as

$$\frac{d}{dt}\Big[t^{\alpha+1}\, e^{-2L/t^\alpha}\, \frac{d}{dt}\mathcal{B}_n^\alpha(t)\Big] = n(n+1)\,\alpha^2\, t^{\alpha-1}\, e^{-2L/t^\alpha}\, \mathcal{B}_n^\alpha(t).$$

By applying integration by parts counter-clockwise along the unit circle, performing some calculations followed by interchanging the role of n and m one can show that the set $\{\mathcal{B}_0^\alpha, \mathcal{B}_1^\alpha, \ldots\}$ constitutes an orthogonal system with regard to the weighting

function $w_{L,\alpha}(t) \equiv t^{\alpha-1}\, e^{-2L/t^{\alpha}}$. Thus, we have

$$\frac{1}{2\pi i}\int_{U} \mathcal{B}_n^{\alpha}(t)\, \mathcal{B}_m^{\alpha}(t)\, w_{L,\alpha}(t)dt = \frac{2L(-1)^{n+1}\,\delta_{nm}}{\alpha(2n+1)}.$$

3.3 The convergence of Bessel solution

In the following we will obtain an approximate solution of (1.2) by a truncated Bessel series in the form

$$u_{N,\alpha}(t) = \sum_{n=0}^{N} a_n\, \mathcal{B}_n^{\alpha}(t), \quad 0 \le t \le L. \tag{3.6}$$

Here, a_n denote the unknown coefficients have to be computed for $n = 0, 1, \ldots, N$. For easy computation let us write $\mathcal{B}_n^{\alpha}(t)$ in the matrix form

$$\boldsymbol{J}_{\alpha}(t) = \boldsymbol{X}_{\alpha}(t)\, \boldsymbol{B}_N^{t}, \tag{3.7}$$

with

$$\boldsymbol{X}_{\alpha}(t) = \begin{bmatrix} 1 & t^{\alpha} & t^{2\alpha} & \ldots & t^{N\alpha} \end{bmatrix}, \quad \boldsymbol{J}_{\alpha}(t) = \begin{bmatrix} \mathcal{B}_0^{\alpha}(t) & \mathcal{B}_1^{\alpha}(t) & \ldots & \mathcal{B}_N^{\alpha}(t) \end{bmatrix}.$$

Here, $\boldsymbol{B}_N = (\beta_{nk})_{n,k=0}^{N}$ is a lower triangular matrix with entries

$$\beta_{nk} := \begin{cases} b_L(n,k), & n \ge k, \\ 0, & n < k. \end{cases}$$

By using (3.7), the equation (3.6) can be written as

$$u_{N,\alpha}(t) = \boldsymbol{J}_{\alpha}(t)\, \boldsymbol{A}_N = \boldsymbol{X}_{\alpha}(t)\, \boldsymbol{B}_N^{t}\, \boldsymbol{A}_N, \tag{3.8}$$

being $\boldsymbol{A}_N = [a_0 \quad a_1 \quad \ldots \quad a_N]^t$ the vector of unknown coefficients.

Let us mention some results on the convergence of fractional Bessel functions. A convergence theorem in the L_2 weighted norm was given in [17]. Indeed, the following theorem shows that by increasing the the number of basis functions N, the approximation solution $u_{N,\alpha}(t)$ converges to $u(t)$ exponentially.

Theorem 3.1 *Let* $\mathbb{B}_N^{\alpha} = Span\langle \mathcal{B}_0^{\alpha}(t), \mathcal{B}_1^{\alpha}(t), \ldots, \mathcal{B}_N^{\alpha}(t)\rangle$ *and* $D^{(k\alpha)}u(t) \in C[0, L]$ *for* $k = 0, 1, \ldots, N+1$. *If* $u_{N,\alpha}(t) = \boldsymbol{J}_{\alpha}(t)\, \boldsymbol{A}_N$ *denotes the best approximation to u from* $\mathbb{B}_N^{\alpha}$, *then the error bound is given by*

$$\|u(t) - u_{N,\alpha}(t)\|_{w_{L,\alpha}} \le \frac{e^{\frac{-1}{L^{\alpha-1}}}\, L^{N\alpha} M_{\infty}^{\alpha}}{\Gamma(N\alpha+\alpha+1)} \sqrt{\frac{L^{3\alpha}}{(2N+3)\alpha}},$$

where $M_{\infty}^{\alpha} := \sup_{t\in[0,L]} |D^{((N+1)\alpha)}u(t)|$.

Proof The proof is similar to those are given in [17,29].

Here and from a different perspective, we give an error bound of the presented method in the L_∞ norm. Below, our approach is based on generalized Taylor's formula [30].

Theorem 3.2 *We assume that $u(t)$ and $u_{N,\alpha}(t) = \boldsymbol{J}_\alpha(t)\,\boldsymbol{A}_N$ are the exact solution and the Bessel polynomial solutions with N-th degree of equation (1.2) for $0 \le t \le L$. Let $u_{N,\alpha}^{Mac}(t) = \boldsymbol{X}_\alpha(t)\,\widehat{\boldsymbol{A}_N}$ be the expansion of the generalized Maclaurin series by (αN)-th degree of $u_{N,\alpha}(t)$. Then, an upper bound for the error of the Bessel polynomial solution $u_{N,\alpha}(t)$ is expressed as follows*

$$\begin{aligned}&\|u(t) - u_{N,\alpha}(t)\|_\infty \\ &\le \frac{L^{\alpha(N+1)}}{\Gamma(\alpha(N+1)+1)}\|u^{(\alpha(N+1))}(\nu_t)\|_\infty + C_\alpha^N\Big(\|\widehat{\boldsymbol{A}_N}\|_\infty + \|\boldsymbol{B}_N^t\|_\infty\|\boldsymbol{A}_N\|_\infty\Big),\end{aligned} \tag{3.9}$$

where $\boldsymbol{X}_\alpha(t)$, $\boldsymbol{J}_\alpha(t)$, and the matrix $\boldsymbol{B}_N$ are as defined in (3.7). Also $C_\alpha^N := \max\{L^{N\alpha}, 1\}$ and $\nu_t \in (0, L)$ is a constant.

Proof We first add and subtract the Maclaurine expansion $u_{N,\alpha}^{Mac}(t)$ by N-th degree into the error $\|u(t) - u_{N,\alpha}\|_\infty$. By utilizing the triangle inequality, we get the following expression

$$\begin{aligned}\|u(t) - u_{N,\alpha}(t)\|_\infty &= \|u(t) - u_{N,\alpha}^{Mac}(t) + u_{N,\alpha}^{Mac}(t) - u_{N,\alpha}(t)\|_\infty \\ &\le \|u(t) - u_{N,\alpha}^{Mac}(t)\|_\infty + \|u_{N,\alpha}^{Mac}(t) - u_{N,\alpha}(t)\|_\infty.\end{aligned} \tag{3.10}$$

By using (3.8), we can express the Bessel polynomial solution $u_{N,\alpha}(t) = \boldsymbol{J}_\alpha(t)\,\boldsymbol{A}_N$ in the matrix form $u_{N,\alpha}(t) = \boldsymbol{X}_\alpha(t)\,\boldsymbol{B}_N^t\,\boldsymbol{A}_N$. Using the fact that the expansion of the Maclaurin series of $u(t)$ by αN-th degree is $u_{N,\alpha}^{Mac}(t) = \boldsymbol{X}_\alpha(t)\,\widehat{\boldsymbol{A}_N}$, we may write

$$\begin{aligned}\|u_{N,\alpha}^{Mac}(t) - u_{N,\alpha}(t)\|_\infty &= \|\boldsymbol{X}_\alpha(t)\,\widehat{\boldsymbol{A}_N} - \boldsymbol{X}_\alpha(t)\,\boldsymbol{B}_N^t\,\boldsymbol{A}_N\|_\infty \\ &\le \|\boldsymbol{X}_\alpha(t)\Big(\widehat{\boldsymbol{A}_N} - \boldsymbol{B}_N^t\boldsymbol{A}_N\Big)\|_\infty \\ &\le \|\boldsymbol{X}_\alpha(t)\|_\infty\Big(\|\widehat{\boldsymbol{A}_N}\|_\infty + \|\boldsymbol{B}_N^t\|_\infty\|\boldsymbol{A}_N\|_\infty\Big), \quad 0 \le t \le L.\end{aligned} \tag{3.11}$$

As $0 \le t \le L$, one can immediately conclude that $\|\boldsymbol{X}_\alpha(t)\|_\infty \le \max\{L^{N\alpha}, 1\} =: C_\alpha^N$. Consequently, we find that

$$\|u_{N,\alpha}^{Mac}(t) - u_{N,\alpha}(t)\|_\infty \le C_\alpha^N\Big(\|\widehat{\boldsymbol{A}_N}\|_\infty + \|\boldsymbol{B}_N^t\|_\infty\|\boldsymbol{A}_N\|_\infty\Big). \tag{3.12}$$

On the other hand, it is known that the reminder term of the Maclaurin series $u_{N,\alpha}^{Mac}(t)$ by (αN)-th degree is bounded by [30]

$$\frac{t^{\alpha(N+1)}}{\Gamma(\alpha(N+1)+1)}\|u^{(\alpha(N+1))}(v_t)\|_\infty, \quad 0 \le t \le L,$$

where $v_t \in (0, L)$. Therefore, we obtain

$$\|u(t) - u_{N,\alpha}^{Mac}(t)\|_\infty \le \frac{L^{\alpha(N+1)}}{\Gamma(\alpha(N+1)+1)}\|u^{(\alpha(N+1))}(v_t)\|_\infty. \tag{3.13}$$

Thus, the proof is completed by combining relations (3.10), (3.12), and (3.13).

Finally, the following collocation points in $0 < t \le L$ will be used throughout the paper. They defined as

$$t_\ell = L_0 + \ell\frac{L - L_0}{N}, \quad \ell = 0, 1, \ldots, N, \tag{3.14}$$

where $0 < L_0 \ll L$ is a real number.

4 The methodology

Let us represent the solution $u(t)$ of Bagley-Torvik equation (1.2) as a linear combination of $(N + 1)$-terms Bessel functions denoted by $u_{N,\alpha}(t)$ on $[0, L]$. According to (3.8), we get

$$u(t) \cong u_{N,\alpha}(t) = \boldsymbol{X}_\alpha(t)\,\boldsymbol{B}_N^t\,\boldsymbol{A}_N. \tag{4.1}$$

We now substitute the collocation points (3.14) into (4.1) to get the following set of matrix equations

$$u_{N,\alpha}(t_\ell) = \boldsymbol{X}_\alpha(t_\ell)\,\boldsymbol{B}_N^t\,\boldsymbol{A}_N, \quad \ell = 0, 1, \ldots, N.$$

In an equivalent compact representation form, the preceding equations can be written as

$$\boldsymbol{U} = \boldsymbol{X}\,\boldsymbol{B}_N^t\,\boldsymbol{A}_N, \tag{4.2}$$

where

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_\alpha(t_0) \\ \boldsymbol{X}_\alpha(t_1) \\ \vdots \\ \boldsymbol{X}_\alpha(t_N) \end{bmatrix}, \quad \boldsymbol{U} = \begin{bmatrix} u_{N,\alpha}(t_0) \\ u_{N,\alpha}(t_1) \\ \vdots \\ u_{N,\alpha}(t_N) \end{bmatrix}.$$

Next, we calculate the q-th order fractional derivative of $u_{N,\alpha}(t)$ in (4.1)

$$D^{(q)}u_{N,\alpha}(t) = D^{(q)}\,\boldsymbol{X}_\alpha(t)\,\boldsymbol{B}_N^t\,\boldsymbol{A}_N. \tag{4.3}$$

Applying the properties (2.2) and (2.3), we are able to compute $D^{(q)}\,\boldsymbol{X}_\alpha(t)$ easily, which is written as follows (see also Algorithm 4.1)

$$\boldsymbol{X}_\alpha^{(q)}(t) := D^{(q)}\,\boldsymbol{X}_\alpha(t) = [0 \quad D^{(q)}\,t^\alpha \quad \dots \quad D^{(q)}\,t^{\alpha N}].$$

Similarly, we place the collocation points (3.14) into the relation (4.3) to arrive at

$$D^{(q)}u_{N,\alpha}(t_\ell) = \boldsymbol{X}_\alpha^{(q)}(t_\ell)\,\boldsymbol{B}_N^t\,\boldsymbol{A}_N, \quad \ell = 0, 1 \dots, N.$$

Consequently, the fractional derivative may be expressed in the matrix form as

$$\boldsymbol{U}^{(q)} = \boldsymbol{X}^{(q)}\,\boldsymbol{B}_N^t\,\boldsymbol{A}_N, \tag{4.4}$$

where

$$\boldsymbol{U}^{(q)} = \begin{bmatrix} D^{(q)}u_{N,\alpha}(t_0) \\ D^{(q)}u_{N,\alpha}(t_1) \\ \vdots \\ D^{(q)}u_{N,\alpha}(t_N) \end{bmatrix}, \quad \boldsymbol{X}^{(q)} = \begin{bmatrix} \boldsymbol{X}_\alpha^{(q)}(t_0) \\ \boldsymbol{X}_\alpha^{(q)}(t_1) \\ \vdots \\ \boldsymbol{X}_\alpha^{(q)}(t_N) \end{bmatrix}.$$

The further aim is to tackle the second-order derivative term $D^2\,u(t)$ in (1.2). Following the same approach used in [6] to find a relationship between $\boldsymbol{X}_\alpha(t)$ and its second derivative to get

$$\frac{d^2}{dt^2}\boldsymbol{X}_\alpha(t) = \boldsymbol{X}_\alpha(t)\,\boldsymbol{T}(t)\,\boldsymbol{S}_\alpha, \tag{4.5}$$

with $\boldsymbol{T}(t) = [1/t^2]$ and

$$\boldsymbol{S}_\alpha = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ 0 & \alpha(\alpha-1) & 0 & \dots & 0 \\ \vdots & \vdots & 2\alpha(2\alpha-1) & \vdots & \vdots \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \dots & N\alpha(N\alpha-1) \end{bmatrix}_{(N+1)\times(N+1)}.$$

Thus, by means of (4.1) and (4.5) we obtain an approximation for $D^2\,u(t)$ as follows

$$D^2\,u(t) \approx \frac{d^2}{dt^2}u_{N,\alpha}(t) = \boldsymbol{X}_\alpha(t)\,\boldsymbol{T}(t)\,\boldsymbol{S}_\alpha\,\boldsymbol{B}_N^t\,\boldsymbol{A}_N. \tag{4.6}$$

If we substitute the collocation points (3.14) into (4.6), we may represent the resulting equations in the matrix form

$$\boldsymbol{U}^2 = \boldsymbol{X}\,\boldsymbol{T}\,\boldsymbol{S}_\alpha\,\boldsymbol{B}_N^t\,\boldsymbol{A}_N, \tag{4.7}$$

where

$$\boldsymbol{U}^2 = \begin{bmatrix} u''_{N,\alpha}(t_0) \\ u''_{N,\alpha}(t_1) \\ \vdots \\ u''_{N,\alpha}(t_N) \end{bmatrix}, \quad \boldsymbol{T} = \begin{bmatrix} \boldsymbol{T}(t_0) & 0 & \dots & 0 \\ 0 & \boldsymbol{T}(t_1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \boldsymbol{T}(t_N) \end{bmatrix}.$$

Now, we collect all terms to define the collocation scheme based on the Bessel polynomials. Let us turn to the original equation (1.2) and substitute the collocation points to obtain

$$P_2(t_\ell)\, D^2\, u(t_\ell) + P_1(t_\ell)\, D^{(q)}\, u(t_\ell) + P_0(t_\ell)\, u(t_\ell) = F(t_\ell), \quad \ell = 0, 1, \dots, N.$$

In a matrix representation, we can rewrite the above equations a

$$\boldsymbol{P}_2\, \boldsymbol{U}^2 + \boldsymbol{P}_1\, \boldsymbol{U}^{(q)} + \boldsymbol{P}_0\, \boldsymbol{U} = \boldsymbol{F}, \tag{4.8}$$

where the involved coefficient matrices $\boldsymbol{P}_k, k = 0, 1, 2$ and the right-hand side vector $\boldsymbol{F}$ are

$$\boldsymbol{P}_k = \begin{bmatrix} P_k(t_0) & 0 & \dots & 0 \\ 0 & P_k(t_1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & P_k(t_N) \end{bmatrix}_{(N+1)\times(N+1)}, \quad \boldsymbol{F} = \begin{bmatrix} F(t_0) \\ F(t_1) \\ \vdots \\ F(t_N) \end{bmatrix}_{(N+1)\times 1}.$$

Substituting the relations (4.2), (4.4), and (4.7) into (4.8), we get the following fundamental matrix equation

$$\boldsymbol{W}\, \boldsymbol{A}_N = \boldsymbol{F}, \quad \text{or} \quad [\boldsymbol{W}; \boldsymbol{F}], \tag{4.9}$$

where

$$\boldsymbol{W} := \left(\boldsymbol{P}_2\, \boldsymbol{X}\, \boldsymbol{T}\, \boldsymbol{S}_\alpha + \boldsymbol{P}_1\, \boldsymbol{X}^{(q)} + \boldsymbol{P}_0\, \boldsymbol{X} \right) \boldsymbol{B}_N^t.$$

Obviously, (4.8) is a linear matrix equation in terms of unknowns Bessel coefficients $a_n, n = 0, 1, \dots, N$ have to be found.

Now, we will implement the initial or boundary conditions (1.3) to convert them into a matrix form. Considering first the initial condition $u(0) = u_0$. By tending $t \to 0$ in (4.1), the following matrix representation form is obtained

$$\widetilde{\boldsymbol{U}}_0\, \boldsymbol{A}_N = u_0, \qquad \widetilde{\boldsymbol{U}}_0 := \boldsymbol{X}_\alpha(0)\, \boldsymbol{B}_N^t = [\tilde{u}_{00} \quad \tilde{u}_{01} \quad \dots \quad \tilde{u}_{0N}].$$

Similarly, for the end condition $u(L) = u_1$, the following matrix expression is obtained

$$\widetilde{\boldsymbol{U}}_L\, \boldsymbol{A}_N = u_1, \qquad \widetilde{\boldsymbol{U}}_L := \boldsymbol{X}_\alpha(L)\, \boldsymbol{B}_N^t = [\tilde{u}_{L0} \quad \tilde{u}_{L1} \quad \dots \quad \tilde{u}_{LN}].$$

For the second initial condition $u'(0) = u_1$, we require to compute the first-order derivative $\frac{d}{dt}\boldsymbol{X}_\alpha(t)$, which clearly depend on N, α. This task can be obtained using the property (2.3). For instance, by selecting $\alpha = 1/3$ and $N = 6$ we have

$$\boldsymbol{X}_{\frac{1}{3}}(t) = \begin{bmatrix} 1 & t^{1/3} & t^{2/3} & t & t^{4/3} & t^{5/3} & t^2 \end{bmatrix}.$$

Differentiation once with respect to t yields

$$\frac{d}{dt}\boldsymbol{X}_{\frac{1}{3}}(t) = \begin{bmatrix} 0 & 0 & 0 & 1 & \frac{4}{3}t^{1/3} & \frac{5}{3}t^{2/3} & 2t \end{bmatrix}.$$

Now, by differentiating one times in the relation (4.1), defining

$$\boldsymbol{X}_\alpha^{(1)}(t) := \frac{d}{dt}\boldsymbol{X}_\alpha(t),$$

and tending $t \to 0$, we conclude that

$$\widetilde{\boldsymbol{U}}_1\, \boldsymbol{A}_N = u_1, \qquad \widetilde{\boldsymbol{U}}_1 := \boldsymbol{X}_\alpha^{(1)}(0)\, \boldsymbol{B}_N^t = [\tilde{u}_{10} \quad \tilde{u}_{11} \quad \ldots \quad \tilde{u}_{1N}].$$

A procedure for computing the $\boldsymbol{X}_\alpha^{(1)}(t)$ by means of (2.3) is presented in Algorithm 4.1. Clearly, the complexity of this algorithm not exceeds $\mathcal{O}(N)$.

Remark 4.1 We emphasize that the calculation of $\boldsymbol{X}_\alpha^{(q)}(t)$ in (4.3) can be performed via calling to Algorithm 4.1 using $\mu = q$. Similarly, one can also compute $\frac{d^2}{dt^2}\boldsymbol{X}_\alpha(t)$ by invoking this Algorithm successively with $\mu = 1$ rather than utilizing the matrix representation as in (4.5). It can be easily verified that the cost of computing $\frac{d^2}{dt^2}\boldsymbol{X}_\alpha(t)$ in (4.5) is about $\mathcal{O}((N+1)^2)$. In the computations below, our approach is based on Algorithm 4.1. This further allows us to tend L_0 to zero when using the collocation points (3.14).

For convenience, we substitute the first row of the fundamental matrix $[\boldsymbol{W}; \boldsymbol{F}]$ by the row matrix $[\widetilde{\boldsymbol{U}}_0; u_0]$ and replace its second row by $[\widetilde{\boldsymbol{U}}_1; u_1]$ or $[\widetilde{\boldsymbol{U}}_L; u_1]$. Thus, the following modified system of linear equations is obtained

$$\widetilde{\boldsymbol{W}}\, \boldsymbol{A}_N = \widetilde{\boldsymbol{F}}, \quad \text{or} \quad [\widetilde{\boldsymbol{W}}; \widetilde{\boldsymbol{F}}]. \tag{4.10}$$

Therefore, by solving the above linear system of equations we are able to acquire the unknown Bessel coefficients in (4.1) straightforwardly.

1: **procedure** $[\boldsymbol{X}_{\alpha}^{\mu}]$= compute_DqX($\mu, \alpha, N$)
2: $\boldsymbol{X}_{\alpha}^{\mu}[1] := 0$;
3: **for** $j := 1, \ldots, N$ **do**
4: **if** $(j\alpha - \mu < 0)$ **then**
5: $\boldsymbol{X}_{\alpha}^{\mu}[j+1] := 0$;
6: **else**
7: **if** $((j\alpha < \lceil \mu \rceil)$ && $(j\alpha - \lfloor j\alpha \rfloor == 0))$ **then**
8: $\boldsymbol{X}_{\alpha}^{\mu}[j+1] := 0$;
9: **else**
10: $\boldsymbol{X}_{\alpha}^{\mu}[j+1] := \dfrac{\Gamma(j\alpha+1)}{\Gamma(j\alpha+1-\mu)} t^{j\alpha-\mu}$;
11: **end if**
12: **end if**
13: **end for**
14: **end;**

Algorithm 4.1: The computation of μ-derivative of the vector $\boldsymbol{X}_{\alpha}(t)$.

5 Numerical and experimental results

To demonstrate the advantages of the proposed new Bessel-collocation method, various practical test examples including both initial and boundary value problems are solved. The numerical results are compared with existing exact solutions and also with the outcomes of other available numerical models. All the experiments were conducted using MATLAB R2017a on a laptop computer with a 1.8 GHz Intel Core $i7$ CPU and 1 GB of memory.

Test problem 5.1 As the first experiment, we consider [9,13]

$$D^2 u(t) + D^{(\frac{1}{2})} u(t) + u(t) = 2 + t^2 + \frac{8}{3\sqrt{\pi}} t\sqrt{t},$$

with the initial conditions $u(0) = 0$ and $u'(0) = 0$. One can easily see that the exact solution is $u(t) = t^2$.

To start, we employ $N = 2$ and $L = 1$. Considering (3.6), we calculate the unknown coefficients a_0, a_1, and a_2. We use the collocation points $t_0 = 0, t_1 = \frac{1}{2}, t_3 = 1$ and take $\alpha = 1$. By taking $q = \frac{1}{2}$, the following vector and matrices related to the matrix equation (4.10) are obtained

$$\boldsymbol{X}^{(\frac{1}{2})} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{679}{851} & \frac{1383}{2600} \\ 0 & \frac{167}{148} & \frac{167}{111} \end{bmatrix}, \quad \boldsymbol{X}^{(2)} = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 2 \\ 0 & 0 & 2 \end{bmatrix}, \quad \boldsymbol{X} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{4} \\ 1 & 1 & 1 \end{bmatrix},$$

$$\boldsymbol{B} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 3 & 3 \end{bmatrix}, \quad \boldsymbol{F} = \begin{bmatrix} 2 \\ \frac{1939}{697} \\ \frac{500}{111} \end{bmatrix}, \quad [\widetilde{\boldsymbol{W}}; \widetilde{\boldsymbol{F}}] = \begin{bmatrix} 1 & 1 & 1 & ; & 0 \\ 1 & \frac{4019}{1749} & \frac{13769}{1040} & ; & \frac{3}{2} \\ 0 & 1 & 3 & ; & 0 \end{bmatrix}.$$

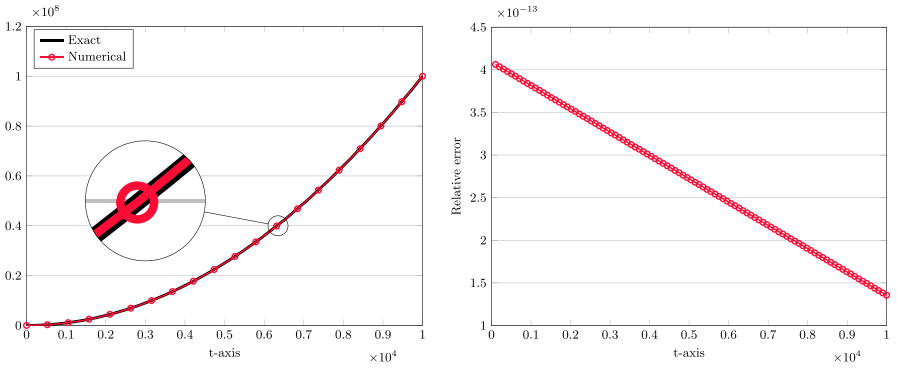

Fig. 2 The graphs of numerical and exact results (left) and the relative error (right) using $N = 3$ and $L = 10000$ in Test Problem 5.1

The coefficients matrix $\boldsymbol{A}_2 = [a_0 \quad a_1 \quad a_2]^t$ will be known after solving the linear system $\widetilde{\boldsymbol{W}}\,\boldsymbol{A}_2 = \widetilde{\boldsymbol{F}}$. This gives us

$$\boldsymbol{A}_2 = [2/3 \quad -1 \quad 1/3]^t.$$

By putting the determined coefficients into $u_{2,1}(t)$ we get the approximate solution

$$u_{2,1}(t) = \begin{bmatrix} 1 & 1+t & 3t^2+3t+1 \end{bmatrix} \boldsymbol{A}_2 = t^2,$$

which is the desired exact solution. For long computation, we take $N = 3$ and $L = 10^4$. In this case, the approximation solution for $0 \le t \le L$ is

$$u_{3,1}(t) = 2.7351675 \times 10^{-17}\, t^3 + 1.0000000000004090479\, t^2 + 2.79049656 \times 10^{-104}\, t.$$

Fig. 2 shows this approximation along with the exact solution. Moreover, the corresponding relative error is visualized on the left plot.

Test problem 5.2 Second test problem is devoted to the following boundary value problem

$$D^{(\frac{3}{2})}\, u(t) + u(t) = t(t-L) + \frac{4\sqrt{t}}{\sqrt{\pi}}.$$

The boundary conditions are $u(0) = 0$, $u(L) = 0$. A direct calculation indicates that the exact solution is given by $u(t) = t^2 - tL$. This example with $L = 1$ is considered in [5,6].

To start, we use $N = 2$ and $\alpha = 1$. When $L = 1$, the same collocation points as in Test Problem 5.1 will be utilized. Applying the proposed scheme for obtaining the unknown coefficients a_0, a_1, a_2 in $u_{2,1}(t)$, yields the following linear system of

equations

$$\begin{cases} a_0 + a_1 + a_2 = 0, \\ a_0 + \frac{3}{2}a_1 + (\frac{12}{\sqrt{2\pi}} + \frac{13}{4})a_2 = -\frac{1}{4} + \frac{4}{\sqrt{2\pi}} \\ a_0 + 2a_2 + 7a_2 = 0, \end{cases}$$

After solving the system we get

$$a_0 = \frac{5}{3}, \quad a_1 = -2, \quad a_2 = \frac{1}{3}.$$

Thus, we obtain the following approximation solution

$$u_{2,1}(t) = \begin{bmatrix} 1 & 1+t & 1+3t+3t^2 \end{bmatrix} \boldsymbol{A}_2 = t^2,$$

which is evidently the exact desired solution. However, to compare our results with traditional Bessel-collocation scheme [6] we use $N = 5$ and $\alpha = 1/2$ to get

$$\begin{aligned} u_{5,\frac{1}{2}}(t) = {} & 0.99999999999998726\, t^2 - 1.000000000000021472\, t + 2.52718 \times 10^{-14}\, t^{1/2} \\ & + 3.06688 \times 10^{-16}\, t^{3/2} + 8.63218 \times 10^{-15}\, t^{5/2} - 2.55478 \times 10^{-108}. \end{aligned}$$

By computing the maximum absolute errors via

$$\mathcal{E}_\infty := \max_{0 \le t \le L} e_{N,\alpha}(t) := \max_{0 \le t \le L} |u(t) - u_{N,\alpha}(t)|,$$

we find that for these values of N, α we achieve that $\mathcal{E}_\infty = 6.7299 \times 10^{-15}$ (at 11 grid points). On the other hand, the corresponding error obtained based on the old Bessel functions is $\mathcal{E}_\infty = 2.2368 \times 10^{-14}$ due to [6]. Further comparisons are reported in Table 1 for $L = 1$ and different N, α. The maximum absolute errors for the larger $L = 10^j$, $j = 1, 2, 4$ are also presented in this table. It can be seen that our approximation procedure is more accurate and particularly with less computational efforts.

The next example indicates that utilizing the Bessel functions of fractional-order is considerably more efficient than integer-order.

Test problem 5.3 Let us take the following initial value problem as the third test case [13]

$$D^2 u(t) + D^{(\frac{3}{2})} u(t) + u(t) = t^{\frac{5}{2}} + \frac{15}{4}\sqrt{t} + \frac{15\sqrt{\pi}}{8} t.$$

The initial conditions are $u(0) = 0$, $u'(0) = 0$ and the corresponding exact solution takes the form $u(t) = t^2\sqrt{t}$.

First, we set $\alpha = 1/2$ and $N = 5$. For $L = 1$, the approximated solution $u_{5,\frac{1}{2}}(t)$ becomes

$$u_{5,\frac{1}{2}}(t) = 8.34666 \times 10^{-14}\, t + 1.03269 \times 10^{-14}\, t^2 - 5.32245 \times 10^{-110}\, t^{1/2}$$

Table 1 Comparing the (maximum) absolute errors for $N = 2, 5$, $\alpha = 1, 1/2$, and various L in Test Problem 5.2

t	$L = 1$			$\mathcal{E}_\infty$		
	New Bessel		Old Bessel [6]	New Bessel		
	$N = 2, \alpha = 1$	$N = 5, \alpha = \frac{1}{2}$	$N = 5, \alpha = \frac{1}{2}$	L	$N = 2, \alpha = 1$	$N = 5, \alpha = \frac{1}{2}$
0.2	0	6.67973_{-15}	2.2037_{-14}	10	2.22559_{-15}	2.59961_{-14}
0.4	0	6.30730_{-15}	2.0808_{-14}	10^2	4.82697_{-14}	4.57319_{-13}
0.6	0	4.65589_{-15}	1.5371_{-14}	10^3	2.73614_{-12}	9.82938_{-11}
0.8	0	2.43416_{-15}	8.0494_{-15}	10^4	3.89749_{-10}	7.61879_{-09}

$$- 1.02655 \times 10^{-14}\, t^{3/2} + 0.99999999999998567319\, t^{5/2} - 2.32857 \times 10^{-110}.$$

However, there are some other options for N, α to obtain an accurate result as above. For instance, by employing $N = 3$, $\alpha = 5/6$ we find the coefficients $a_0, a_2 = -1/3, a_1 = 3/5, a_3 = 1/15$ for $u_{3,\frac{5}{6}}(t)$. Therefore, we get the following approximations for $0 \le t \le 1$

$$\begin{aligned} u_{3,\frac{5}{6}}(t) &= \begin{bmatrix} 1 & 1 + t^{5/6} & 1 + 3t^{5/6} + 3t^{5/3} & 1 + 6t^{5/6} + 15t^{5/3} + 15t^{5/2} \end{bmatrix} \boldsymbol{A}_3 \\ &= 0.99999999999999982576\, t^{5/2} + 2.4222 \times 10^{-16}\, t^{5/3} + 1.7032 \times 10^{-108}\, t^{5/6} \end{aligned}$$

Next goal is to see the benefit of utilizing the fractional-order basis functions rather than integer-order counterparts for $\alpha = 1$. In Table 2 we tabulate the results of errors calculated in the L_∞ and L_2 norms for different $N = 2, 3, 4, 5, 10$ and $\alpha = 5/4, 1, 5/6, 5/8, 1/2$. The L_2-error norm is denoted by $\mathcal{E}_2$ and is defined by

$$\mathcal{E}_2 := \Big(\int_0^L \big(u(t) - u_{N,\alpha}(t) \big)^2 \, dt \Big)^{1/2}.$$

Note that, for calculating $\mathcal{E}_\infty$ we have divided $[0, L]$ into ten subintervals. In addition, the CPU times measured in second required to solve the fundamental matrix equation (4.10) are also reported in Table 2.

For large computations related to Test Problem 5.3, we fix $N = 2$ and $\alpha = 5/4$. The numerical results $u_{N,\alpha}(t)$ using $L = 10, 100$, and $L = 1000$ evaluated at points $t = L/5, 2L/5, L/5, 4L/5, L$ are shown in Table 3. The corresponding errors in the L_2 norm as well as the required CPU times are also reported on the last rows of Table 3. The graphs of absolute errors using $N = 3, \alpha = 5/6$ on the intervals $0 \le t \le L$ with $L = 10^3$ and $L = 10^4$ are depicted in Fig. 3. The corresponding L_2 error norms are $\mathcal{E}_2 = 1.28896 \times 10^{-9}$ and 4.47452×10^{-6} respectively.

Test problem 5.4 Consider the boundary value problem

$$\begin{aligned} &D^2 u(t) + \frac{8}{17} D^{(\frac{3}{2})} u(t) + \frac{13}{51} u(t) \\ &\quad = \frac{1}{89250\sqrt{\pi} L^5} (96\sqrt{t}\, F_1(t) + 7\sqrt{\pi}\, F_2(t)), \quad 0 < t < L, \end{aligned}$$

with boundary conditions $u(0) = 0, u(L) = 0$. One can see that the exact solution is $u(t) = \frac{27}{125L} t - \frac{339}{250L^2} t^2 + \frac{76}{25L^3} t^3 - \frac{29}{10L^4} t^4 + \frac{1}{L^5} t^5$. Here, two functions $F_1(t)$ and $F_2(t)$ are

$$\begin{aligned} F_1(t) &= 8000t^3 - 16240Lt^2 + 10640L^2 t - 2373L^3, \\ F_2(t) &= 3250t^5 - 9425Lt^4 + (255000 + 9880L^2)t^3 - (443700 + 4407L^2)Lt^2 \\ &\quad + (232560 + 702L^2)L^2 t - 34578L^3. \end{aligned}$$

When $L = 1$, this model problem is considered in [4,6,7,10,11].

Table 2 Comparison of L_∞, L_2 error norms and CPU times (in second) for test problem (5.3) using $L = 1$

	$N = 10, \alpha = 1$	$N = 5, \alpha = \frac{1}{2}$	$N = 4, \alpha = \frac{5}{8}$	$N = 3, \alpha = \frac{5}{6}$	$N = 2, \alpha = \frac{5}{4}$
$\mathcal{E}_\infty$	2.3996_{-03}	6.9201_{-14}	4.3499_{-15}	6.9322_{-17}	5.5844_{-17}
$\mathcal{E}_2$	1.5661_{-03}	4.2094_{-14}	2.3141_{-15}	4.7563_{-17}	2.2798_{-17}
CPU time	0.0872	0.0692	0.0637	0.0752	0.0717

Table 3 Numerical solutions of Test Problem 5.3 using $N = 2, \alpha = 5/4$, and various L

t	$L = 10$	$L = 100$	$L = 1000$
$\frac{L}{5}$	5.6568542494923805	1788.8543819998317	565685.42494923797
$\frac{2L}{5}$	32.000000000000002	10119.288512538813	3199999.9999999997
$\frac{3L}{5}$	88.181630740194416	27885.480092693400	8818163.0740194404
$\frac{4L}{5}$	181.01933598375618	57243.340223994613	18101933.598375615
L	316.22776601683795	99999.999999999995	31622776.601683791
$\mathcal{E}_2$	2.02432×10^{-14}	1.91318×10^{-11}	3.34036×10^{-08}
CPU time	0.14074	0.14143	0.14247

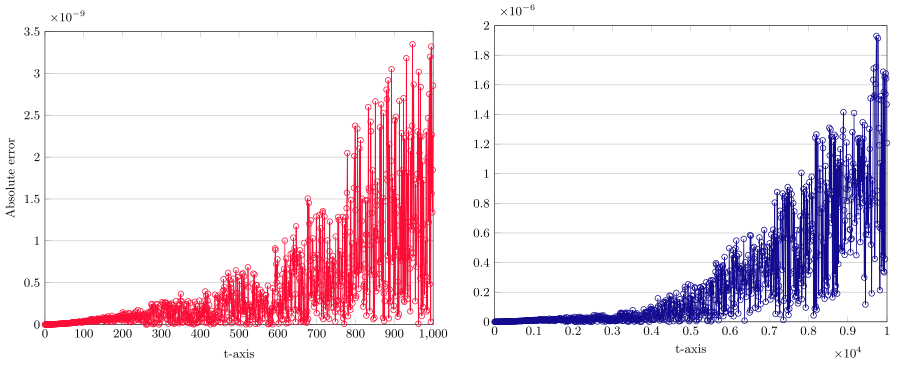

Fig. 3 Absolute errors for $L = 10^3$ (left) and $L = 10^4$ (right) with $N = 3, \alpha = 5/6$ in Test Problem 5.3

Let start computations with $N = 5$ and set $\alpha, L = 1$. The approximate solution $u_{5,1}(t)$ of Test Problem 5.4 using new Bessel functions is obtained for $t \in [0, 1]$ as

$$\begin{aligned} u_{5,1}(t) = {} & 0.99999999999999925931\, t^5 - 2.8999999999999981346\, t^4 \\ & + 3.0399999999999986223\, t^3 - 1.3559999999999995687\, t^2 \\ & + 0.21599999999999982165\, t, \end{aligned}$$

which is exact solution up to machine epsilon. In this case, the corresponding errors achieved as

$$\mathcal{E}_\infty = 3.826596 \times 10^{-16}, \quad \mathcal{E}_2 = 4.48753 \times 10^{-17},$$

while the maximum absolute error using the old Bessel polynomials using $N = 8$ and $\alpha = 1/2$ is reported as $\mathcal{E}_\infty = 1.1074 \times 10^{-2}$ in [6]. Of course, we are suprised that why the author in [6] utilized these values of N and α, which are obviously not reasonable choices. In fact, using the traditional Bessel polynomials with $N = 5$ and $\alpha = 1$ we find the same approximation as our method.

Our next aim is to validate and compare our results more with existing numerical schemes. To this end, the absolute errors based on Haar wavelet [4] and sinc [10]

Table 4 Comparing of absolute errors for Test Problem 5.4 using $N = 5, \alpha = 1$, $L = 1$, and various $t \in [0, L]$

t	New Bessel ($N = 5, \alpha = 1$) Numerical	$e_{5,1}(t)$	Haar wavelet [4] $m = 256$	Sinc [10] $N = 64$
0.1	0.01079999999999998528	1.47206_{-17}	3.89916_{-6}	3.92_{-9}
0.2	0.00895999999999997331	2.66913_{-17}	4.17115_{-6}	4.59_{-9}
0.3	0.00377999999999996143	3.85737_{-17}	3.94253_{-6}	4.16_{-9}
0.4	−0.00000000000000005033	5.18095_{-18}	3.37304_{-6}	4.12_{-9}
0.5	−0.00100000000000006011	6.01128_{-17}	2.60917_{-6}	3.87_{-9}
0.6	−0.00000000000000006515	4.58704_{-17}	1.78827_{-6}	3.82_{-9}
0.7	0.00125999999999993735	9.51057_{-18}	1.04097_{-6}	4.54_{-9}
0.8	0.00127999999999994935	5.70189_{-17}	4.92027_{-7}	4.78_{-9}
0.9	−0.00000000000000002895	1.09824_{-16}	2.61094_{-7}	3.16_{-9}

Table 5 Comparing the L_∞, L_2 error norms for Test Problem (5.4) using $N = 5, \alpha = 1$, and different L

	$L = 10$	$L = 100$	$L = 1000$	$L = 10000$
$\mathcal{E}_\infty$	8.0946797_{-16}	1.0722311_{-15}	9.4551759_{-16}	6.7976870_{-16}
$\mathcal{E}_2$	4.2026154_{-16}	1.4015331_{-15}	8.6380776_{-16}	4.9938022_{-15}
CPU time	0.15909	0.16163	0.16213	0.16531

operational matrices are presented in Table 4. It can be observed from Table 4 that our experimental results are in excellent alignment with the exact true solutions.

Since the exact solution of Test Problem 5.4 is a five-degree polynomials, using $N = 5$ and $\alpha = 1$ are also sufficient for larger values of L. Fixing these values, the approximated solutions for $L = 10, 100$ take the following forms respectively

$$\begin{aligned} u_{5,1}^{10}(t) &= 0.00001000000000000000002614\, t^5 - 0.00029000000000000020278\, t^4 \\ &\quad + 0.00304000000000000024978\, t^3 - 0.01355999999999999301\, t^2 \\ &\quad + 0.02159999999999999919864\, t + 2.12897992000407536 \times 10^{-108}, \\ u_{5,1}^{100}(t) &= 0.00000000009999999999998662\, t^5 - 0.00000002899999999999730196\, t^4 \\ &\quad + 0.00000303999999999999812868\, t^3 - 0.00013559999999999943298\, t^2 \\ &\quad + 0.00215999999999999944118\, t - 7.4514297200142637598 \times 10^{-108}. \end{aligned}$$

The corresponding error norms $\mathcal{E}_\infty$ and $\mathcal{E}_2$ for $L = 10, 100$ as well as $L = 1000$, $L = 10000$ are shown in Table 5. It should be note that the computations of $\mathcal{E}_\infty$ are obtained at 500 grid points for each L. Besides, the CPU times needed for these values of L are further presented in Table 5. Finally, in Fig. 4 the plots of numerical and exact solutions for $L = 1000$ are given. The corresponding absolute error are also visualized in Fig. 4, right plot.

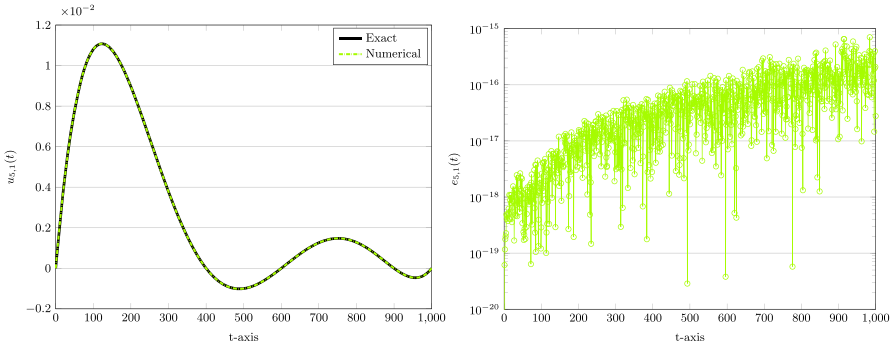

Fig. 4 Graphs of numerical and exact results (left) and the absolute error (right) using $N = 5, \alpha = 1$, and $L = 1000$ in Test Problem 5.4

Test problem 5.5 Let us consider a three-points boundary-value problem of fractional Bagley-Torvik equation. This model with variable coefficients is

$$D^2 u(t) - 5t\sqrt{\pi}\, D^{(\frac{1}{2})} u(t) + 16\sqrt{t}\, u(t) = 6t,$$

with initial conditions $u(0)+2u(\frac{L}{2}) = \frac{L^3}{4}, u(L) = L^3$. The exact solution is $u(t) = t^3$. For $L = 1/5$, the model problem studied in [32,33].

We first consider $\alpha = 1$ and $N = 3$. For $L = 1/5$, the approximative solution for $0 \le t \le L$ is obtained as

$$\begin{aligned} u_{3,1}(t) = {} & 0.99999999999999999433\, t^3 + 1.109796 \times 10^{-18}\, t^2 \\ & + 3.438436 \times 10^{-20}\, t - 5.911134 \times 10^{-21}. \end{aligned}$$

The numerical results correspond the above approximation are summarized in Table 6. These results are given at some points $t \in [0, L]$. The related absolute errors $e_{N,\alpha}(t)$ are also reported in Table 6. For comparison's sake, the results obtained by the Galerkin and least square methods based on boundary characteristic orthogonal polynomials (BCOPs) [32] and the method based on Fredholm integral equations (FIEs) of the second kind [33] are also displayed in Table 6. As one can see from Table 6 that there is a strong level of agreement in our approximative solutions compared to the corresponding exact solutions.

Now, we examine the effect of using a larger value of L on the computations. We fix $N = 3$ and $\alpha = 1$. For instance, with $L = 1$ and $L = 10$ the approximations obtained via the novel Bessel-collocation scheme are

$$\begin{aligned} u^1_{3,1}(t) = {} & 0.99999999999999972383\, t^3 + 3.37495 \times 10^{-16}\, t^2 \\ & - 4.21388 \times 10^{-17}\, t - 1.918894513603734 \times 10^{-17}, \quad t \in [0, 1], \\ u^{10}_{3,1}(t) = {} & 0.99999999999999748188\, t^3 + 4.51842 \times 10^{-14}\, t^2 \\ & - 2.18561 \times 10^{-13}\, t + 1.853092782769985 \times 10^{-13}, \quad t \in [0, 10]. \end{aligned}$$

Table 6 The comparison of numerical solutions and the related absolute errors for $N = 3, \alpha = 1$ and various $t \in [0, L]$, $L = 1/5$ in Test Problem 5.5

t	New Bessel		BCOPs [33]		FIEs [32]	
	$u_{3,1}(t)$	$e_{3,1}(t)$	Galerkin	Least square	$(m, n) = (4, 2)$	Exact
0.00	0.000_{+0}	5.9111_{-21}	0.00	0.00	8.1180_{-11}	0.000_{+0}
0.02	8.000_{-6}	4.8249_{-21}	8.000960_{-6}	7.999399_{-6}	2.5205_{-10}	8.000_{-6}
0.04	6.400_{-5}	3.1229_{-21}	64.001745_{-6}	64.000183_{-6}	1.2628_{-09}	6.400_{-5}
0.06	2.160_{-4}	1.0775_{-21}	216.001933_{-6}	216.001022_{-6}	1.1225_{-09}	2.160_{-4}
0.08	5.120_{-4}	1.0394_{-21}	512.001337_{-6}	512.001076_{-6}	3.7779_{-10}	5.120_{-4}
0.10	1.000_{-3}	2.9556_{-21}	1_{-3}	1_{-3}	4.0590_{-11}	1.000_{-3}
0.12	1.728_{-3}	4.3988_{-21}	1727.998200_{-6}	1727.99794_{-6}	3.4848_{-09}	1.728_{-3}
0.14	2.744_{-3}	5.0970_{-21}	–	–	1.4666_{-09}	2.744_{-3}
0.16	4.096_{-3}	4.7781_{-21}	–	–	3.3408_{-09}	4.096_{-3}
0.18	5.832_{-3}	3.1698_{-21}	–	–	3.5001_{-09}	5.832_{-3}

Table 7 Absolute errors for Test Problem 5.5 using $N = 3, \alpha = 1$, $L = 1, 10, 100, 1000$, and some $t \in [0, L]$

t	$L = 1$	$L = 10$	$L = 100$	$L = 1000$
$\frac{L}{10}$	2.0304_{-17}	9.4147_{-15}	9.5333_{-12}	1.3143_{-09}
$\frac{L}{5}$	1.6326_{-17}	9.1220_{-14}	1.1195_{-10}	5.8066_{-08}
$\frac{3L}{10}$	8.9126_{-18}	1.3170_{-13}	1.6029_{-10}	8.0717_{-08}
$\frac{2L}{5}$	2.8001_{-19}	1.2715_{-13}	1.5392_{-10}	7.6033_{-08}
$\frac{L}{2}$	9.5945_{-18}	9.2655_{-14}	1.1131_{-10}	5.3409_{-08}
$\frac{3L}{5}$	1.7374_{-17}	4.3338_{-14}	5.0878_{-11}	2.2240_{-08}
$\frac{7L}{10}$	2.1961_{-17}	5.6940_{-15}	8.9128_{-12}	8.0803_{-09}
$\frac{4L}{5}$	2.1699_{-17}	3.9334_{-14}	4.9620_{-11}	2.8157_{-08}
$\frac{9L}{10}$	1.4931_{-17}	4.2472_{-14}	5.2798_{-11}	2.8595_{-08}
$\mathcal{E}_2$	1.5898×10^{-17}	1.8531×10^{-13}	2.2261×10^{-10}	1.0682×10^{-07}

A comparison of absolute errors using different $L = 1, 10, 100, 1000$ are shown in Table 7. These errors at some points in $[0, L]$ are computed. Moreover, in the last row, the corresponding L_2 error norms are reported.

In the next experiments, some realistic model problems with oscillatory behaviour are studied.

Test problem 5.6 We consider the fractional relaxation-osilation equation [1,11,35]

$$D^{(q)}\, u(t) + u(t) = F(t), \quad q = \frac{3}{2}, \quad 0 \le t \le L,$$

with the initial condition $u(0) = 0$ and $u'(0) = 0$. The exact solution is given by $u(t) = \int_0^t G(t-s)F(s)ds$. Here, $G(t) = t^{q-1}E_{q,q}(-t^q)$ and $E_{a,b}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(ak+b)}$ is the two parameter Mittag-Leffler function. Below, two types of $F(t) = 1$ and $F(t) = t\, e^{-t}$ will be first used with $L = 20$. Hence, we use $F(t) = t^{\frac{5}{2}} + \frac{15\, t}{8\sqrt{\pi}}$ for comparison purpose. In this case, the aforementioned exact solution is reduced to explicit form $u(t) = \sqrt{t^5}$.

We take $N = 20$ and $\alpha = 1$ for the computations. Note that in this example the roots of Chebyshev polynomials are utilized as collocation points due to their slightly more accuracy. Let us first consider $F(t) = 1$. In Table 8, the numerical results, the exact solutions, and the related absolute errors at some points $t \in [0, L]$, $L = 20$ are reported. The plots of approximate solution (coloured curve) along with exact solution (solid line) are visualized in Fig. 5.

In the second case, we employ $F(t) = te^{-t}$. The presentation of numerical solutions with coressponding absolute errors are carried out in Table 9 and further is shown in Fig. 6. Additionally, a comparison has been made in Table 9 with the results obtained by p-order fractional linear multiple step methods (p-HOFLMSM) [34]. The value of p used is three with other parameters $n = 100$ and $h = 0.2$. Referring to Fig. 6 and

Table 8 Numerical and exact solutions along with absolute errors with $F(t) = 1$ in Test Problem 5.6 for $N = 20$ and $\alpha = 1$

t	$u_{20,1}(t)$	Exact	$e_{20,1}(t)$
0.80	0.4266918	0.4595831	3.2891234_{-2}
2.40	1.2448989	1.2535658	8.6668464_{-3}
4.00	1.2094895	1.2028715	6.6179332_{-3}
5.60	1.0123145	1.0091265	3.1879510_{-3}
7.20	9.7735801	9.7781119	4.5317888_{-4}
8.80	1.0060444	1.0068499	8.0551041_{-4}
10.4	1.0152587	1.0150447	2.1394822_{-4}
12.0	1.0086875	1.0086134	7.4098862_{-5}
13.6	1.0046824	1.0045499	1.3252496_{-4}
15.2	1.0041732	1.0041914	1.8186027_{-5}
16.8	1.0040492	1.0042058	1.5654784_{-4}
18.4	1.0036454	1.0036980	5.2646940_{-5}
20.0	1.0040605	1.0031463	9.1414010_{-4}

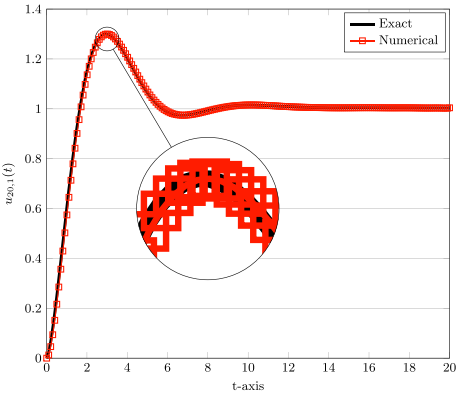

Fig. 5 Plot of numerical approximation for $F(t) = 1$ and $N = 20$ in Test Problem 5.6

Table 9, it is clearly observed that our approximation solutions are in good alignment with the exact true solutions by using of novel Bessel polynomials.

In the last and third case, we consider $F(t) = t^{\frac{5}{2}} + \frac{15t}{8\sqrt{\pi}}$. We further set $L = 1$ and $N = 2, 5$. In Table 10, we present the pointwise absolute errors $e_{N,\alpha}(t)$ for diverse values of $\alpha = \frac{5}{4}, \frac{1}{2}, \alpha = 1$ and some values of $t \in [0, L]$. Since the exact solution has fractional power, selecting fractional α give rise to the most accurate results clearly. To validate our results, the outcomes of the achieved pointwise error E obtained via Fermat tau operational matrix method [11] using $M = 9$ are also reported in Table 10.

6 Concluding remarks

In this research work, we have developed an effective matrix method based upon generalized Bessel polynomials to find an approximate solution of the well-known

Table 9 Comparison of approximated solutions and related absolute errors in Test Problem 5.6 for N, $L = 20$, $\alpha = 1$, $F(t) = te^{-t}$, and different $t \in [0, L]$

t	New Bessel		3-HOFLMSM [34]		
	$u_{20,1}(t)$	$e_{20,1}(t)$	Numerical	Error	Exact [34]
0.80	1.0032955_{-1}	1.12498_{-3}	9.52875892_{-2}	3.91698327_{-3}	9.92045725_{-2}
2.40	3.8526751_{-1}	2.84175_{-4}	3.84651536_{-1}	3.31797377_{-4}	3.84983333_{-1}
4.00	2.0099338_{-1}	2.32662_{-4}	2.01589390_{-1}	3.63344850_{-4}	2.01226046_{-1}
5.60	-1.2459924_{-2}	1.02407_{-4}	-1.20968485_{-2}	2.60668883_{-4}	-1.23575173_{-2}
7.20	-4.0943173_{-2}	3.17363_{-6}	-4.09010633_{-2}	4.52828845_{-5}	-4.09463462_{-2}
8.80	-7.1474761_{-3}	4.08428_{-5}	-7.20316073_{-3}	1.48418475_{-5}	-7.18831888_{-3}
10.4	4.4881833_{-3}	1.91551_{-5}	4.52776157_{-3}	2.04231665_{-5}	4.50733840_{-3}
12.0	4.3529718_{-4}	7.72247_{-6}	4.73026337_{-4}	4.54516234_{-5}	4.27574713_{-4}
13.6	-2.0459496_{-3}	1.19979_{-5}	-1.99206193_{-3}	4.18897439_{-5}	-2.03395168_{-3}
15.2	-1.2546984_{-3}	1.72339_{-6}	-1.22253777_{-3}	3.38840349_{-5}	-1.25642181_{-3}
16.8	-3.8306143_{-4}	2.02946_{-5}	-3.70895123_{-4}	3.24609452_{-5}	-4.03356068_{-4}
18.4	-2.6163963_{-4}	8.69192_{-6}	-2.36528966_{-4}	3.38025783_{-5}	-2.70331545_{-4}
20.0	-4.2950679_{-4}	1.10739_{-4}	-2.84607314_{-4}	3.41605531_{-5}	-3.18767867_{-4}

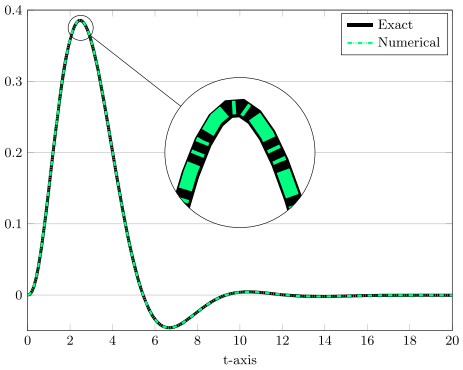

Fig. 6 Numerical and exact solutions using $F(t) = te^{-t}$ for $N, L = 20, \alpha = 1$ in Test Problem 5.6

Table 10 Comparison of pointwise absolute errors for test problem (5.6) for $N = 2, 5, \alpha = \frac{5}{4}, \frac{1}{2}, 1$, $F(t) = t^{\frac{5}{2}} + \frac{15t}{8\sqrt{\pi}}$, and different $t \in [0, L]$. $L = 1$

	$t = 0.0$	$t = 0.2$	$t = 0.4$	$t = 0.6$	$t = 0.8$	$t = 1.0$
$e_{2,\frac{5}{4}}(t)$	0	3.68_{-18}	2.08_{-17}	5.74_{-17}	1.18_{-16}	2.06_{-16}
$e_{5,\frac{1}{2}}(t)$	0	1.07_{-15}	2.04_{-15}	2.86_{-15}	3.51_{-15}	3.96_{-15}
$e_{5,1}(t)$	0	1.06_{-03}	1.24_{-03}	1.50_{-03}	1.52_{-03}	1.79_{-03}
E [11]	0	2.28_{-09}	1.21_{-08}	8.60_{-09}	1.18_{-08}	2.45_{-09}

fractional-order Bagley-Torvik differential equations over a long time domain. Representing all unknowns in the matrix forms and with the help of collocation points, the presented approach transforms the original equations into an algebraic system of linear equations. The error and convergence results on the spectral Bessel solutions are discussed. A fast algorithm (Algorithm 4.1) with linear complexity for computing the μ-derivative the of involved basis functions in the vector form is proposed. Numerous numerical examples have been conducted to testify the practicability of the Bessel-collocation approach when applied to the fractional Bagley-Torvik equation. A comparison of the computational results with analytical solutions and existing numerical models indicates that the presented algorithm is an improvement over other well-established methods in terms of accuracy and efficiency.

References

1. Podlubny, I.: Fractional Differential Equations. Academic Press, New York (1999)
2. Kilbas, A.A., Srivastava, H.M., Trujillo, J.J.: Theory and Applications of Fractional Differential Equations. Elsevier, San Diego (2006)
3. Bagley, R.L., Torvik, P.J.: On the appearance of the fractional derivative in the behaviour of real materials. J. Appl. Mech. **51**(2), 294–298 (1984)
4. Raja, M.A.Z., Khan, J.A., Qureshi, I.M.: Solution of fractional order system of Bagley-Torvik equation using evolutionary computational intelligence. Math. Probl. Eng. **2011**, 675075 (2011)
5. Srivastava, H.M., Shah, F.A., Abass, R.: An application of the Gegenbauer wavelet method for the numerical solution of the fractional Bagley-Torvik equation. Russ. J. Math. Phys. **26**(1), 77–93 (2019)

6. Yüzbasi, S.: Numerical solution of the Bagley-Torvik equation by the Bessel collocation method. Math. Meth. Appl. Sci. **36**(3), 300–312 (2012)
7. Mashayekhi, S., Razzaghi, M.: Numerical solution of the fractional Bagley-Torvik equation by using hybrid functions approximation. Math. Meth. Appl. Sci. **39**(3), 353–365 (2016)
8. Karaaslan, M.F., Celiker, F., Kurulay, M.: Approximate solution of the Bagley-Torvik equation by hybridizable discontinuous Galerkin methods. Appl. Math. Comput. **285**, 51–58 (2016)
9. Izadi, M., Negar, M.R.: Local discontinuous Galerkin approximations to fractional Bagley-Torvik equation. Math. Meth. Appl. Sci. **43**(7), 4978–4813 (2020)
10. Azizi, M.-R., Khani, A.: Sinc operational matrix method for solving the Bagley-Torvik equation. Comput. Methods Differ. Equ. **5**(1), 56–66 (2017)
11. Youssri, Y.H.: A new operational matrix of Caputo fractional derivatives of Fermat polynomials: an application for solving the Bagley-Torvik equation. Adv. Differ. Equ. **2017**, Article ID 73 (2017)
12. Atta, A.G., Moatimid, G.M., Youssri, Y.H.: Generalized Fibonacci operational tau algorithm for fractional Bagley-Torvik equation. Progr. Fract. Differ. Appl. **6**(3), 1–10 (2020)
13. Rahimkhani, P., Ordokhani, Y.: Application of Müntz-Legendre polynomials for solving the Bagley-Torvik equation in a large interval. SeMA J. **75**(3), 517–533 (2018)
14. Arqub, O.A., Maayah, B.: Solutions of Bagley-Torvik and Painlevé equations of fractional order using iterative reproducing kernel algorithm with error estimates. Neural Comput. Applic. **29**(5), 1465–1479 (2018)
15. Yüzbasi, S., Karacayır, M.: A Galerkin-type fractional approach for solutions of Bagley-Torvik equations. Comput. Model. Eng. Sci. **123**(3), 941–956 (2020)
16. Krall, H.L., Frink, O.: A new class of orthogonal polynomials: The Bessel polynomials. Trans. Amer. Math. Soc. **65**, 100–115 (1949)
17. Izadi, M., Cattani, C.: Generalized Bessel polynomial for multi-order fractional differential equations. Symmetry **12**(8), Article ID 1260 (2020)
18. Izadi, M., Srivastava, H.M.: An efficient approximation technique applied to a non-linear Lane-Emden pantograph delay differential model. Appl. Math. Comput. **401**, Article ID 126123 (2021)
19. Izadi, M.: Numerical approximation of Hunter-Saxton equation by an efficient accurate approach on long time domains. U.P.B. Sci. Bull. Series A **83**(1), 291–300 (2021)
20. Izadi, M., Yüzbasi, S., Noeiaghdam, S.: Approximating solutions of non-linear Troesch's problem via an efficient quasi-linearization Bessel approach. Mathematics **9**(16), Article ID 1841 (2021)
21. Izadi, M.: A combined approximation method for nonlinear foam drainage equation, Sci. Iran. (2021) 10.24200/sci.2021.56571.4792
22. Abd-Elhameed, W.M., Youssri, Y.H.: Generalized Lucas polynomial sequence approach for fractional differential equations. Nonlinear Dyn. **89**(2), 1341–1355 (2017)
23. Izadi, M.: A comparative study of two Legendre-collocation schemes applied to fractional logistic equation. Int. J. Appl. Comput. Math. **6**(3), Article ID 71 (2020)
24. Yüzbasi, S.: A collocation method for numerical solutions of fractional-order logistic population model. Int. J. Biomath. **9**(2), 1650031–45 (2016)
25. M. Izadi, Srivastava, H.M.: A novel matrix technique for multi-order pantograph differential equations of fractional order, Proc. Roy. Soc. London Ser. A: Math. Phys. Engrg. Sci. **477** (2021), Article ID 2021031, 1-21
26. Yüzbasi, S.: A numerical approximation for Volterra's population growth model with fractional order. Appl. Math. Model. **37**, 3216–3227 (2013)
27. Izadi, M., Srivastava, H.M.: Numerical approximations to the nonlinear fractional-order Logistic population model with fractional-order Bessel and Legendre bases. Chaos Solitons Fract. **145**, Article ID 110779 (2021)
28. Sabermahani, S., Ordokhani, Y., Yousefi, S.A.: Numerical approach based on fractional-order Lagrange polynomials for solving a class of fractional differential equations. Comput. Appl. Math. **37**(3), 3846–3868 (2018)
29. Parand, K., Delkhosh, M.: Solving Volterra's population growth model of arbitrary order using the generalized fractional order of the Chebyshev functions. Ricerche Mat. **65**, 307–328 (2016)
30. Odibat, Z., Shawagfeh, N.T.: Generalized Taylor's formula. Appl. Math. Comput. **186**(1), 286–293 (2007)
31. Wei, H.M., Zhong, X.C., Huang, Q.A.: Uniqueness and approximation of solution for fractional Bagley-Torvik equations with variable coefficients. Int. J. Comput. Math. **94**(1), 1541–1561 (2016)

32. Huang, Q.A., Zhong, X.C., Guo, B.L.: Approximate solution of Bagley-Torvik equations with variable coefficients and three-point boundary-value conditions. Int. J. Appl. Comput. Math. **2**, 327–347 (2016)
33. Jena, R.M., Chakraverty, S.: Boundary characteristic orthogonal polynomials-based Galerkin and least square methods for solving Bagley-Torvik equations. In: Recent Trends in Wave Mechanics and Vibrations 2020 (pp. 327-342). Springer, Singapore
34. Lin, R., Liu, F.: Fractional high order methods for the nonlinear fractional ordinary differential equation. Nonlinear Anal. **66**(4), 856–869 (2007)
35. Izadi, M.: A computational algorithm for simulating fractional order relaxation-oscillation equation. SeMA J. (2021). https://doi.org/10.1007/s40324-021-00266-x

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.