Mathematics and Computers in Simulation 238 (2025) 479–496

Contents lists available at ScienceDirect

Mathematics and Computers in Simulation

journal homepage: www.elsevier.com/locate/matcom

Original articles

Application of cubic B-splines for second order fractional Sturm–Liouville problems

Arezu Aghazadeh [a], Mehrdad Lakestani [a,b,c,*]

[a] Faculty of Mathematics, Statistics and Computer Science, University of Tabriz, Tabriz, Iran
[b] Research Center of Performance and Productivity Analysis, Istinye University, Istanbul, Turkiye
[c] Jadara University Research Center, Jadara University, Irbid, Jordan

ARTICLE INFO

MSC:
49N10
65D07
65R10
65L60

Keywords:
Sturm–Liouville eigenvalue problem
Caputo fractional derivative
Cubic B-spline

ABSTRACT

This paper presents a novel numerical method for determining the eigenvalues and eigenfunctions of a fractional second-order Sturm–Liouville problem, a model that represents a broad class of fractional-order differential equations commonly encountered in mathematical physics and engineering. The proposed approach combines cubic B-spline functions with the Galerkin method, transforming the problem into a system of algebraic equations. Additionally, operational matrices for fractional-order integrals and derivatives are constructed. The technique's effectiveness and precision are illustrated through its application to various example problems.

1. Introduction

In this study, we examine the fractional Sturm–Liouville problem

$$ {}_{0}^{C}D_{t}^{\alpha} y(t) + (\lambda r(t) - q(t))y(t) = 0, \quad 0 \le t \le 1, \tag{1.1} $$

where $r(t)$ and $q(t)$ are continuous real-valued functions, with $r(t) \neq 0$ and $1 < \alpha \le 2$. In this equation, ${}_{0}^{C}D_{t}^{\alpha} y(t)$ represents the Caputo fractional derivative of $y(t)$. The boundary conditions are given by

$$ \begin{cases} ay(0) + by'(0) = 0, \\ cy(1) + dy'(1) = 0, \end{cases} \tag{1.2} $$

where the constants a, b, c, and d are not all zero simultaneously, ensuring that $a^2 + b^2 \neq 0$ and $c^2 + d^2 \neq 0$.

The classical integer-order Sturm–Liouville problem (when $\alpha = 2$) is well-established in mathematical physics and has been widely studied [1–3]. However, the fractional counterpart of this equation has garnered significant interest from researchers over the past decade.

The existence of solutions for the fractional Sturm–Liouville problem was first explored in [4,5], where eigenvalues for certain classes of problems were found to correspond to the roots of specific combinations of orthogonal functions.

After developing fractional derivative operators during last decade, such as Riemann–Liouville, Grunwald–Letnikov, Caputo, Caputo–Fabrizio [6], Atangana–Baleanu, many researchers began to consider fractional order differential equations as suitable tools for modeling natural phenomenons. Due to the nature of the Caputo fractional derivative in better compliance with initial conditions

* Corresponding author at: Faculty of Mathematics, Statistics and Computer Science, University of Tabriz, Tabriz, Iran.
E-mail address: lakestani@tabrizu.ac.ir (M. Lakestani).

https://doi.org/10.1016/j.matcom.2025.06.030
Received 1 October 2024; Received in revised form 23 April 2025; Accepted 26 June 2025

Available online 7 July 2025

in natural problems, this fractional derivative is more popular than other fractional derivatives. For this reason, we use Caputo's fractional derivative in this research.

Numerous semi-analytical and numerical methods have since been employed to solve Sturm–Liouville problems. For example, Al-Mdallal et al. utilized the Adomian Decomposition Method (ADM) [7,8], while Abbasbandy et al. applied the Homotopy Analysis Method (HAM) [9]. Wavelet-based approaches were introduced by Nematy [10] and Shi [11], while other methods such as Fourier series [12], Laplace transforms [13], Radial Basis Functions (RBF) [14], and the Galerkin finite element method [15] have also been developed. More recently, the fractional series method was used by Al-Mdallal et al. to tackle fourth-order fractional Sturm–Liouville problems, where the eigenfunctions were expressed in terms of Mittag-Leffler functions [16]. Kashkari et al. provided a theoretical and numerical analysis of nonsingular fractional second-order Sturm–Liouville problems, implementing the fractional Legendre Tau method [17].

Calculating eigenvalues with high index for Sturm–Liouville problems usually involves greater errors. Authors in [18], applied a new efficient method to the Sturm–Liouville problems subject to Rubin boundary conditions in which the high-indexed eigenvalues are computed more accurately. A numerical scheme for the regular fractional SturmLiouville problem containing the Prabhakar fractional derivatives was studied in [19]. For further works, we also refer to [20,21].

Due to the rich geometric properties of Spline functions, they have proven to be highly effective tools for curve approximation. As a result, various numerical methods utilizing B-spline interpolation have been developed for solving fractional differential equations. For instance, cubic trigonometric B-spline functions were employed to solve the generalized nonlinear time-fractional Klein–Gordon equation with the Caputo operator in [22]. In [23], cubic B-spline functions were used to approximate the time-fractional Allen–Cahn equation, while an efficient technique based on cubic B-splines was proposed for solving the time-fractional advection–diffusion equation in [24]. Additionally, extended cubic B-splines were applied to the time-fractional telegraph equation in [25], and a cubic B-spline collocation method was introduced for solving linear and nonlinear fractional integro-differential equations in [26]. Edrisi et al. [27] also employed the B-spline collocation method to address fractional optimal control problems. For further applications of B-splines, we refer to [28–32]. However, cubic B-Splines have not yet been applied to solve fractional Sturm–Liouville problems. The flexibility of cubic B-Spline bases for approximating functions can be effective in solving Sturm–Liouville problems. Moreover, achieving high accuracy with orthogonal polynomials, such as Chebyshev and Legendre polynomials, often requires incorporating additional terms, which leads to an increase in the polynomial degree. Long ago, Walter Gautschi [33] demonstrated that the condition number of such a conversion grows as $(1+\sqrt{2})^N$ where N is the degree of the polynomials. In the present method, the degree of the polynomials is fixed at 3, ensuring that the method remains stable.

In this paper, we aim to approximate the eigenvalues and corresponding eigenfunctions of the Sturm–Liouville problem (1.1) with mixed boundary conditions (1.2) using a numerical method based on cubic B-splines.

The remainder of this paper is structured as follows: In Section 2, after introducing polynomial cubic B-splines and their key properties, we compute the corresponding operational matrices for product, fractional integrals, and derivatives. Section 3 discusses the implementation of cubic B-spline bases for the fractional Sturm–Liouville problem. Section 4 briefly addresses the method's convergence. Numerical results are provided in Section 5, followed by a brief discussion in Section 6.

2. B-splines and their properties

The mth-order B-spline $N_m(t)$ is defined on the knot sequence $\{\dots,-2,-1,0,1,2,\dots\}$ and represents a polynomial of order m (degree $m-1$) between two consecutive knots. Starting with $N_1(t)=\chi_{[0,1]}(t)$, the characteristic function on $[0,1]$, the mth-order B-spline is constructed recursively, as described in [34]:

$$N_m(t)=(N_{m-1}*N_1)(t)=\int_{-\infty}^{\infty}N_{m-1}(t-\tau)N_1(\tau)d\tau=\int_0^1 N_{m-1}(\tau)d\tau. \tag{2.1}$$

A recursive formula for $N_m(t)$ (for $m\geq 2$) is provided in [35]:

$$N_m(t)=\frac{t}{m-1}N_{m-1}(t)+\frac{m-t}{m-1}N_{m-1}(t-1). \tag{2.2}$$

It is clear that the support of $N_1(t)$ is $[0,1]$. According to Eq. (2.2), the support of $N_2(t)$ is $[0,2]$. Following this pattern, we conclude that the support of $N_m(t)$ is $[0,m]$.

The explicit form of the cubic B-spline ($N_4(t)$) was given in [34,35] as follows:

$$N_4(t)=\begin{cases}1/6t^3, & 0\leq t<1,\\ -1/2t^3+2t^2-2t+2/3, & 1\leq t<2,\\ 1/3t^3-4t^2+10t-22/3, & 2\leq t<3,\\ -1/6t^3+2t^2-8t+32/3, & 3\leq t<4,\\ 0, & \text{elsewhere}.\end{cases} \tag{2.3}$$

By scaling and shifting the function $N_4(t)$, the family of cubic B-splines $N_{M,k}(t)$ $(M, k \in Z)$ is defined as

$$N_{M,k}(t) = N_4(2^M t - k), \qquad M, k \in Z,$$

where M and k are the scaling and shifting factors, respectively.

The support of $N_{M,k}(t)$ is given by

$$B_{M,k} = \text{supp}(N_{M,k}(t)) = [2^{-M}k, 2^{-M}(k+4)], \qquad M, k \in Z.$$

To apply cubic B-splines on the interval $[0,1]$, we use a finite set of functions specifically, those that do not vanish on this interval. For a fixed M, let

$$S_M = \{k : N_{M,k}[2^{-M}k, 2^{-M}(k+4)] \cap [0,1] \neq \emptyset\}, \qquad M \in Z.$$

It is straightforward to show that

$$S_M = \{-3, -2, \ldots, 2^M - 1\}.$$

2.1. Function approximation with cubic B-splines

For a fixed M, we define the cubic B-spline vector of size $2^M + 3$ as follows:

$$\Phi_M = [N_{M,-3}(t), N_{M,-2}(t), \ldots, N_{M,2^M-1}(t)]^T, \tag{2.4}$$

where $N_{M,k}(t)$ $(k = -3, \ldots, 2^M - 1)$ are restricted to the domain $[0,1]$. A function $f(t) \in L^2[0,1]$ can be approximated by cubic B-splines as:

$$f(t) \simeq \sum_{k=-3}^{2^M-1} c_k N_{M,k}(t) = C^T \Phi_M(t), \tag{2.5}$$

where the vector C contains the coefficients:

$$C = [c_{-3}, c_{-2}, c_{-1}, \ldots, c_{2^M-1}]^T. \tag{2.6}$$

To determine the coefficient vector C, we multiply the approximation $f(t) = C^T \Phi_M(t)$ by $\Phi_M^T(t)$ and take the inner product:

$$\left\langle f(t), \Phi_M^T(t) \right\rangle = C^T \left\langle \Phi_M(t), \Phi_M^T(t) \right\rangle,$$

This leads to:

$$C = W^{-1} F, \tag{2.7}$$

where

$$W = \left\langle \Phi_M(t), \Phi_M^T(t) \right\rangle, \tag{2.8}$$

is a symmetric, invertible matrix, and

$$F^T = \left\langle f(t), \Phi_M^T(t) \right\rangle.$$

The elements of W and F are given by:

$$W_{ij} = \left\langle N_{M,i}(t), N_{M,j}(t) \right\rangle, \qquad F_i = \left\langle f(t), N_{M,i}(t) \right\rangle,$$

where $i, j = -3, -2, \ldots, 2^M - 1$. Here, $\langle f, g \rangle$ represents the inner product of f and g on the interval $[0,1]$.

To compute W_{ij}, we have:

$$\begin{aligned}
W_{ij} &= \int_0^1 N_{M,i}(t) N_{M,j}(t) dt \\
&= \int_{\max(0, \frac{i}{2^M}, \frac{j}{2^M})}^{\min(1, \frac{i+4}{2^M}, \frac{j+4}{2^M})} N_{M,i}(t) N_{M,j}(t) dt \\
&= \int_{\max(0, \frac{i}{2^M}, \frac{j}{2^M})}^{\min(1, \frac{i+4}{2^M}, \frac{j+4}{2^M})} N_4(2^M t - i) N_4(2^M t - j) dt
\end{aligned}$$

It should be noted that the integral is computed over the common support of the functions $N_{M,i}(t)$ and $N_{M,j}(t)$, i.e.,

$$\left(\frac{i}{2^M}, \frac{i+4}{2^M}\right) \cap \left(\frac{j}{2^M}, \frac{j+4}{2^M}\right) \cap [0,1].$$

Using the fact that $N_{M,k}(t) = N_4(2^M t - k)$, we apply the change of variable $x = 2^M t$, leading to:

$$W_{ij} = \frac{1}{2^M} \int_{\max(0,i,j)}^{\min(2^M, i+4, j+4)} N_4(x-i) N_4(x-j) dx.$$

The matrix W is a seven-diagonal matrix, with the nonzero elements described as:

$$W = \frac{1}{2^M} \begin{bmatrix} \frac{1}{252} & \frac{43}{1680} & \frac{1}{84} & \frac{1}{5040} & & & & & \\ \frac{43}{1680} & \frac{151}{630} & \frac{59}{280} & \frac{1}{42} & \frac{1}{5040} & & & & \\ \frac{1}{84} & \frac{59}{280} & \frac{599}{1260} & \frac{397}{1680} & \frac{1}{42} & \frac{1}{5040} & & & \\ \frac{1}{5040} & \frac{1}{42} & \frac{397}{1680} & \frac{151}{315} & \frac{397}{1680} & \frac{1}{42} & \frac{1}{5040} & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & \frac{1}{5040} & \frac{1}{42} & \frac{397}{1680} & \frac{151}{315} & \frac{397}{1680} & \frac{1}{42} & \frac{1}{5040} \\ & & & \frac{1}{5040} & \frac{1}{42} & \frac{397}{1680} & \frac{599}{1260} & \frac{59}{280} & \frac{1}{84} \\ & & & & \frac{1}{5040} & \frac{1}{42} & \frac{59}{280} & \frac{151}{630} & \frac{43}{1680} \\ & & & & & \frac{1}{5040} & \frac{1}{84} & \frac{43}{1680} & \frac{1}{252} \end{bmatrix} \tag{2.9}$$

Lemma 2.1. *The matrix W defined in (2.8) is an invertible symmetric matrix.*

Proof. By the definition of matrix W we have

$$W_{ij} = \langle N_{M,i}(t), N_{M,j}(t) \rangle = \langle N_{M,j}(t), N_{M,i}(t) \rangle = W_{ji}, \qquad i,j = -3,-2,\dots,2^M-1.$$

The symmetry of W follows directly from the inner product properties. Given the linear independence of the B-spline basis functions $\{N_{M,k}\}_{k=-3}^{2^M-1}$, any linear combination of the form

$$\sum_{j=-3}^{2^M-1} c_j N_{M,j}(t) = 0,$$

must imply that $c_j = 0$ for all $j = -3,-2,\dots,2^M-1$. Multiplying the above equation by $N_{M,i}(t)$ and integrating over the interval $[0,1]$ gives

$$WC = 0, \tag{2.10}$$

where $C = (c_{-3}, c_{-2}, \dots, c_{2^M-1})^T$ and W is defined in (2.8). Since the system of homogeneous Eqs. (2.10) has only the trivial solution $C = 0$, it follows that W is invertible.

2.2. Operational matrix for function multiplication

In this section, we explain how to approximate the product of two cubic B-spline vectors by constructing an operational matrix. This process is essential for implementing numerical methods that involve B-spline functions.

We begin with the product of two cubic B-spline vectors $\Phi_M(t)$ and its transpose, multiplied by a given vector Y:

$$\Phi_M(t)\Phi_M^T(t)Y = \widetilde{Y}\Phi_M(t), \tag{2.11}$$

where $\widetilde{Y}$ is a transformed vector to be determined. To determine $\widetilde{Y}$, we multiply the equation by $\Phi_M^T(t)$ and integrate over the interval $[0,1]$. This leads to the equation:

$$\widetilde{Y} = R\ W^{-1},$$

where $W = \langle \Phi_M(t), \Phi_M^T(t) \rangle$ is the matrix of inner products of cubic B-splines, and $R = \langle \Phi_M(t)\Phi_M^T(t)Y, \Phi_M^T(t) \rangle$ is a matrix representing the product of three B-splines.

The matrix R is a square matrix of size $(2^M+3) \times (2^M+3)$, and its elements are computed as follows:

$$R_{i,j} = \sum_{k=-3}^{2^M-1} q_{ijk} y_k, \qquad i,j = -3,-2,\dots,2^M-1,$$

where q_{ijk} is the integral of the product of three cubic B-splines:

$$q_{ijk} = \int_0^1 N_{M,i}(t) N_{M,j}(t) N_{M,k}(t) dt, \qquad i,j,k = -3,-2,\dots,2^M-1.$$

The coefficient q_{ijk} represents the integral over the interval $[0,1]$ of the product of three cubic B-splines $N_{M,i}(t)$, $N_{M,j}(t)$, and $N_{M,k}(t)$. Due to the compact support of cubic B-splines, many of these integrals will be zero because the supports of the splines do not overlap. The non-zero integrals occur when the supports of the three B-splines overlap in a small region.

2.3. Operational matrix of integration

In this section, we derive the operational matrix for the integration of cubic B-splines. This matrix is useful for numerical methods involving B-splines where integration is required.

Consider the integral of the cubic B-spline vector $\Phi_M(t)$, which can be expressed as:

$${}_0I_t\Phi_m(t) = \int_0^t \Phi_M(\tau) d\tau \simeq P\ \Phi_M(t), \tag{2.12}$$

where P is a $(2^M+3)\times(2M+3)$ matrix. The approximation uses the matrix P to represent the integral in terms of the B-spline basis functions. To determine P, we multiply both sides of the equation by $\Phi_M^T(t)$ and integrate over the interval $[0,1]$. This simplifies to:

$$P = B\ W^{-1}, \tag{2.13}$$

where B is defined as:

$$B = \langle {}_0I_t\Phi_M(t), \Phi_M^T(t) \rangle .$$

The matrix B is a $(2^M+3) \times (2^M+3)$ matrix with elements given by:

$$B_{ij} = \int_0^1 {}_0I_t N_{M,i}(t) N_{M,j}(t) dt. \tag{2.14}$$

Since the support of ${}_0I_t N_{M,i}(t)$ is $(i/2^M, +\infty)$, we can further simplify this by breaking down the integral:

$$B_{ij} = \int_{\max(0,\frac{i}{2^M},\frac{j}{2^M})}^{\min(1,\frac{j+4}{2^M})} {}_0I_t N_{M,i}(t) N_{M,j}(t) dt. \tag{2.15}$$

Applying the change of variable $x = 2^M t$, the integral becomes:

$$B_{ij} = \frac{1}{2^{2M}} \int_{\max(0,i,j)}^{\min(j+4,2^M)} {}_0I_x N_4(x-i) N_4(x-j) dx, \quad i,j = -3,-2,\dots,2^M-1. \tag{2.16}$$

where ${}_0I_x$ represents the integral of the B-spline $N_4(x-i)$ up to x.

The matrix B can be calculated using a block structure due to the repetitive nature of the B-splines. This reduces computational complexity as many elements in B are repeated.

$$B = \frac{1}{2^{2M}} \times$$
$$\left[\begin{array}{ccccccccccc|ccc}
\frac{1}{1152} & \frac{629}{40320} & \frac{169}{4480} & \frac{1679}{40320} & \frac{1}{24} & \frac{1}{24} & \frac{1}{24} & \frac{1}{24} & \frac{1}{24} & \cdots & \frac{1}{24} & \frac{23}{576} & \frac{1}{48} & \frac{1}{576} \\
\frac{211}{40320} & \frac{1}{8} & \frac{1583}{4032} & \frac{2489}{5040} & \frac{20159}{40320} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \cdots & \frac{1}{2} & \frac{23}{48} & \frac{1}{4} & \frac{1}{48} \\
\frac{89}{40320} & \frac{349}{4032} & \frac{529}{1152} & \frac{34099}{40320} & \frac{4799}{5040} & \frac{38639}{40320} & \frac{23}{24} & \frac{23}{24} & \frac{23}{24} & \cdots & \frac{23}{24} & \frac{529}{576} & \frac{23}{48} & \frac{23}{576} \\
\hline
\frac{1}{40320} & \frac{31}{5040} & \frac{4541}{40320} & \frac{1}{2} & \frac{35779}{40320} & \frac{5009}{5040} & \frac{40319}{40320} & 1 & 1 & \cdots & 1 & \frac{23}{24} & \frac{1}{2} & \frac{1}{24} \\
 & \frac{1}{40320} & \frac{31}{5040} & \frac{4541}{40320} & \frac{1}{2} & \frac{35779}{40320} & \frac{5009}{5040} & \frac{40319}{40320} & 1 & \cdots & 1 & \frac{23}{24} & \frac{1}{2} & \frac{1}{24} \\
 & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \\
 & & & \frac{1}{40320} & \frac{31}{5040} & \frac{4541}{40320} & \frac{1}{2} & \frac{35779}{40320} & \frac{5009}{5040} & \frac{40319}{40320} & 1 & \frac{23}{24} & \frac{1}{2} & \frac{1}{24} \\
 & & & & \frac{1}{40320} & \frac{31}{5040} & \frac{4541}{40320} & \frac{1}{2} & \frac{35779}{40320} & \frac{5009}{5040} & \frac{40319}{40320} & \frac{23}{24} & \frac{1}{2} & \frac{1}{24} \\
 & & & & & \frac{1}{40320} & \frac{31}{5040} & \frac{4541}{40320} & \frac{1}{2} & \frac{35779}{40320} & \frac{5009}{5040} & \frac{38639}{40320} & \frac{1}{2} & \frac{1}{24} \\
 & & & & & & \frac{1}{40320} & \frac{31}{5040} & \frac{4541}{40320} & \frac{1}{2} & \frac{35779}{40320} & \frac{4799}{5040} & \frac{20159}{40320} & \frac{1}{24} \\
 & & & & & & & \frac{1}{40320} & \frac{31}{5040} & \frac{4541}{40320} & \frac{1}{2} & \frac{34099}{40320} & \frac{2489}{5040} & \frac{1679}{40320} \\
 & & & & & & & & \frac{1}{40320} & \frac{31}{5040} & \frac{4541}{40320} & \frac{529}{1152} & \frac{1583}{4032} & \frac{169}{4480} \\
 & & & & & & & & & \frac{1}{40320} & \frac{31}{5040} & \frac{349}{4032} & \frac{1}{8} & \frac{629}{40320} \\
 & & & & & & & & & & \frac{1}{40320} & \frac{89}{40320} & \frac{211}{40320} & \frac{1}{1152}
\end{array}\right] \tag{2.17}$$

As shown in Eq. (2.17), Matrix B has a block structure as follows:

$$B = \frac{1}{2^{2M}} \left[\begin{array}{c|c} B_1 & B_2 \\ \hline B_3 & B_4 \end{array}\right]. \tag{2.18}$$

The components of this matrix are calculated using relation (2.16). In Block B_3, when $|i-j|$ remains constant, the components are repeated.

2.4. Operational matrix for fractional integration

The fractional integral of order α of the cubic B-spline vector $\Phi_M(t)$ can be represented in terms of an operational matrix P^α. This section outlines the derivation of P^α and its computation.

The fractional integral of order α for the cubic B-spline vector $\Phi_M(t)$ is given by:

$${}_0I_t^\alpha \Phi_M(t) = P^\alpha\ \Phi_M(t), \tag{2.19}$$

where P^α is a $(2^M+3)\times(2^M+3)$ matrix. This matrix P^α helps approximate the fractional integral of the B-splines. To determine P^α, multiply both sides of Eq. (2.19) by $\Phi_M^T(t)$ and integrate over $[0,1]$. This simplifies to:

$$P^\alpha = B^\alpha\ W^{-1}, \tag{2.20}$$

where B^α is defined as:

$$B^\alpha = \left\langle {}_0I_t^\alpha \Phi_M(t), \Phi_M^T(t) \right\rangle .$$

The matrix B^α is a $(2^M+3)\times(2^M+3)$ matrix with elements:

$$\begin{aligned} B_{ij}^\alpha &= \int_0^1 {}_0I_t^\alpha N_{M,i}(t) N_{M,j}(t)dt \\ &= \int_{\max(0,\frac{j}{2^M})}^{\min(1,\frac{j+4}{2^M})} {}_0I_t^\alpha N_{M,i}(t) N_{M,j}(t)dt. \end{aligned} \tag{2.21}$$

Using the change of variable $x = 2^M t$, the integral becomes:

$$B^{\alpha}_{ij} = \frac{1}{2^{M(\alpha+1)}} \int_{\max(0,j)}^{\min(j+4,2^M)} {}_0I^{\alpha}_x N_4(x-i)N_4(x-j)dx, \quad i,j = -3,-2,\dots,2^M-1. \tag{2.22}$$

Thus, we arrive at:

$$B^{\alpha} = \frac{1}{2^{M(\alpha+1)}} \times$$

$$\left[\begin{array}{cccccccccc|ccc}
b_{-3,-3} & b_{-3,-2} & b_{-3,-1} & b_{-3,0} & b_{-3,1} & b_{-3,2} & b_{-3,3} & b_{-3,4} & \cdots & b_{-3,2^M-4} & b_{-3,2^M-3} & b_{-3,2^M-2} & b_{-3,2^M-1} \\
b_{-2,-3} & b_{-2,-2} & b_{-2,-1} & b_{-2,0} & b_{-2,1} & b_{-2,2} & b_{-2,3} & b_{-2,4} & \cdots & b_{-2,2^M-4} & b_{-2,2^M-3} & b_{-2,2^M-2} & b_{-2,2^M-1} \\
b_{-1,-3} & b_{-1,-2} & b_{-1,-1} & b_{-1,0} & b_{-1,1} & b_{-1,2} & b_{-1,3} & b_{-1,4} & \cdots & b_{-1,2^M-4} & b_{-1,2^M-3} & b_{-1,2^M-2} & b_{-1,2^M-1} \\
\hline
b'_{-3} & b'_{-2} & b'_{-1} & b'_{0} & b'_{1} & b'_{2} & b'_{3} & b'_{4} & \cdots & b'_{2^M-4} & b_{0,2^M-3} & b_{0,2^M-2} & b_{0,2^M-1} \\
 & b'_{-3} & b'_{-2} & b'_{-1} & b'_{0} & b'_{1} & b'_{2} & b'_{3} & \cdots & b'_{2^M-5} & b_{1,2^M-3} & b_{1,2^M-2} & b_{1,2^M-1} \\
 & & b'_{-3} & b'_{-2} & b'_{-1} & b'_{0} & b'_{1} & b'_{2} & \cdots & b'_{2^M-6} & b_{2,2^M-3} & b_{2,2^M-2} & b_{2,2^M-1} \\
 & & & \ddots & \ddots & \ddots & \ddots & \ddots & & & \vdots & \vdots & \vdots \\
 & & & & b'_{-3} & b'_{-2} & b'_{-1} & b'_{0} & b'_{1} & b'_{2} & b_{2^M-6,2^M-3} & b_{2^M-6,2^M-2} & b_{2^M-6,2^M-1} \\
 & & & & & b'_{-3} & b'_{-2} & b'_{-1} & b'_{0} & b'_{1} & b_{2^M-5,2^M-3} & b_{2^M-5,2^M-2} & b_{2^M-5,2^M-1} \\
 & & & & & & b'_{-3} & b'_{-2} & b'_{-1} & b'_{0} & b_{2^M-4,2^M-3} & b_{2^M-4,2^M-2} & b_{2^M-4,2^M-1} \\
 & & & & & & & b'_{-3} & b'_{-2} & b'_{-1} & b_{2^M-3,2^M-3} & b_{2^M-3,2^M-2} & b_{2^M-3,2^M-1} \\
 & & & & & & & & b'_{-3} & b'_{-2} & b_{2^M-2,2^M-3} & b_{2^M-2,2^M-2} & b_{2^M-2,2^M-1} \\
 & & & & & & & & & b'_{-3} & b_{2^M-1,2^M-3} & b_{2^M-1,2^M-2} & b_{2^M-1,2^M-1}
\end{array}\right] \tag{2.23}$$

Matrix B^{α} has a block structure due to the repetitive nature of the B-splines, which simplifies computation. It can be represented as:

$$B^{\alpha} = \frac{1}{2^{M(\alpha+1)}} \left[\begin{array}{c|c} B^{\alpha}_1 & B^{\alpha}_2 \\ \hline B^{\alpha}_3 & B^{\alpha}_4 \end{array}\right], \tag{2.24}$$

where the block B^{α}_3 is upper triangular matrix and the members parallel to the diameter are repeated. Therefore, only the first row of block B^{α}_3 will be calculated. the blocks B^{α}_1, B^{α}_2 and B^{α}_4 can be computed using the same integral approach, leveraging the fact that elements in these blocks often repeat due to the overlapping nature of the B-splines. By (2.23) the entries in (2.23) are as follows

$$b_{i,j} = \int_{\max(0,j)}^{\min(j+4,2^M)} {}_0I^{\alpha}_x N_4(x-i)N_4(x-j)dx, \quad i,j = -3,-2,\dots,2^M-1,$$

and

$$b'_j = b_{0,j} = \int_{\max(0,j)}^{\min(j+4,2^M)} {}_0I^{\alpha}_x N_4(x)N_4(x-j)dx, \quad j = -3,-2,\dots,2^M-4.$$

3. Application to the fractional Sturm–Liouville problem

In this section, we apply cubic B-spline approximation to the fractional Sturm–Liouville problem (1.1)–(1.2). We use cubic B-splines to approximate the solution of the differential equation with given boundary conditions.

Let us approximate the second derivative $y''(t)$ using cubic B-splines:

$$y''(t) = C^T \Phi_M(t), \tag{3.1}$$

where $\Phi_M(t)$ represents the cubic B-spline vector and C is the vector of unknown coefficients.

Integrating (3.1) twice gives us:

$$y'(t) = y'(0) + C^T P \Phi_M(t), \tag{3.2}$$

$$y(t) = y(0) + y'(0)t + C^T P^2 \Phi_M(t), \tag{3.3}$$

where P represents the operational matrix for integration.

In Eqs. (3.2) and (3.3), the values of $y(0)$ and $y'(0)$ might not be directly provided by the boundary conditions in (1.2). Therefore, we need to determine $y(0)$ and $y'(0)$ using the boundary conditions in (1.2). Based on the values of a, b, c, and d in (1.2), four possible cases arise:

1. $a \neq 0$ and $c \neq 0$.
2. $a \neq 0$ and $d \neq 0$.

3. $b \neq 0$ and $c \neq 0$.
4. $b \neq 0$ and $d \neq 0$.

In the following, we will focus on the first case, as the other three can be handled in a similar manner.

For the first case, where $a \neq 0$ and $c \neq 0$, the boundary conditions in (1.2) provide:

$$y(0) = -\frac{b}{a} y'(0), \qquad y(1) = -\frac{d}{c} y'(1).$$

Substituting $t = 1$ into Eqs. (3.2) and (3.3) results in:

$$\begin{cases} y'(1) = y'(0) + C^T P \Phi_M(1), \\ y(1) = y(0) + y'(0) + C^T P^2 \Phi_M(1). \end{cases} \tag{3.4}$$

This gives:

$$\begin{cases} y'(1) = y'(0) + C^T P \Phi_M(1), \\ -\frac{d}{c} y'(1) = -\frac{b}{a} y'(0) + y'(0) + C^T P^2 \Phi_M(1). \end{cases} \tag{3.5}$$

From this, we can derive:

$$\begin{cases} y'(0) = \dfrac{ac}{bc - ad - ac} C^T \left(P^2 + \dfrac{d}{c} P \right) \Phi_M(1), \\ y(0) = \dfrac{-bc}{bc - ad - ac} C^T \left(P^2 + \dfrac{d}{c} P \right) \Phi_M(1). \end{cases} \tag{3.6}$$

Substituting the expression for $y'(0)$ into (3.3), we obtain:

$$y(t) = C^T \left(P^2 + \frac{d}{c} P \right) \Phi_M(1)\ g(t) + C^T P^2 \Phi_M(t), \tag{3.7}$$

where $g(t) = \dfrac{-bc + act}{bc - ad - ac}$. This formula is also applicable to the third case.

In cases 2 and 4, where $d \neq 0$, following a similar process, we obtain:

$$y(t) = C^T \left(\frac{c}{d} P^2 + P \right) \Phi_M(1)\ \bar{g}(t) + C^T P^2 \Phi_M(t), \tag{3.8}$$

where $\bar{g}(t) = \dfrac{-bd + adt}{bc - ad - ac}$.

Next, we represent $g(t)$ from (3.7) using cubic B-splines as:

$$g(t) = G^T \Phi_M(t),$$

which allows us to rewrite (3.7) as:

$$y(t) = C^T \left(P^2 + \frac{d}{c} P \right) \Phi_M(1)\ G^T \Phi_M(t) + C^T P^2 \Phi_M(t). \tag{3.9}$$

Similarly, in the case where $d \neq 0$, we get:

$$y(t) = C^T \left(\frac{c}{d} P^2 + P \right) \Phi_M(1)\ \bar{G}^T \Phi_M(t) + C^T P^2 \Phi_M(t), \tag{3.10}$$

where $\bar{g}(t) = \bar{G}^T \Phi_M(t)$.

With this setup, we are now ready to apply the cubic B-spline approximation to the Sturm–Liouville eigenvalue problem (1.1)–(1.2). We approximate $y''(t)$ using (3.1) and $y(t)$ using (3.9) or (3.10). The remaining terms in (1.1) can be approximated as follows:

$${}_0^C D_t^\alpha y(t) = {}_0 I_t^{2-\alpha} y''(t) = C^T {}_0 I_t^{2-\alpha} \Phi_M(t) = C^T\ P^{2-\alpha} \Phi_M(t), \tag{3.11}$$

$$r(t) = R^T \Phi_M(t), \qquad q(t) = Q^T \Phi_M(t). \tag{3.12}$$

where $P^{2-\alpha}$ is the operational matrix for fractional integration of order $2 - \alpha$.

We now substitute (3.1), (3.9), (3.11), and (3.12) into (1.1), giving:

$$C^T P_M^{2-\alpha} \Phi_M(t) + C^T \left\{ \left(P^2 + \frac{d}{c} P \right) \Phi_M(1)\ G^T + P^2 \right\} \Phi_M(t) \Phi_M^T(t) (\lambda R - Q) = 0. \tag{3.13}$$

Applying the operational matrix product to this equation yields:

$$C^T P^{2-\alpha} \Phi_M(t) + C^T \left\{ \left(P^2 + \frac{d}{c} P \right) \Phi_M(1)\ G^T + P^2 \right\} (\lambda \tilde{R} - \tilde{Q}) \Phi_M(t) = 0. \tag{3.14}$$

Since the B-spline basis is linearly independent, Eq. (3.14) can be written in matrix form as:

$$\mathbf{A} C = \lambda \mathbf{B} C, \tag{3.15}$$

where:

$$\mathbf{A} = \left[P^{2-\alpha} - \left\{ \left(P^2 + \frac{d}{c} P \right) \Phi_M(1)\, G^T + P^2 \right\} \tilde{Q} \right]^T, \tag{3.16}$$

and:

$$\mathbf{B} = -\left[\left\{ \left(P^2 + \frac{d}{c} P \right) \Phi_M(1)\, G^T + P^2 \right\} \tilde{R} \right]^T. \tag{3.17}$$

It is important to note that in the case $\alpha = 2$, we set $P^{2-\alpha} = I$ in (3.16), where I is the identity matrix of the appropriate size.

The goal of solving the fractional Sturm–Liouville problem (1.1)–(1.2) is to find nontrivial solutions, which is equivalent to identifying the eigenvalues in the generalized eigenvalue problem (3.15). Therefore, we need to solve the generalized eigenvalue problem (3.15). Modern mathematical software packages provide highly efficient tools to accurately compute these eigenvalues.

Once we approximate the eigenvalues, we can then find the corresponding eigenfunctions $y_\lambda(t)$ for each eigenvalue λ. To achieve this, we solve the singular system (3.15). A common approach is to set the coefficient $C_{2^M-1} = c$ (where c is a free parameter) and solve (3.15) to determine the remaining coefficients C_k for $k = -3, -2, \ldots, 2^M - 2$ in terms of c.

The eigenfunction $y_\lambda(t)$ can then be approximated by:

$$y_\lambda(t) \simeq \sum_{k=-3}^{2^M-1} C_k(c) N_{M,k}(t), \tag{3.18}$$

where $N_{M,k}(t)$ are the cubic B-splines.

To uniquely determine the eigenfunction $y_\lambda(t)$, an additional auxiliary condition (separate from the boundary conditions) must be applied. Common choices for such an auxiliary condition include setting $y'_\lambda(0) = 1$ or $y''_\lambda(0) = 1$. This ensures that the eigenfunction is uniquely determined for each eigenvalue λ.

4. Error study

The following theorem presents the approximation error for cubic splines.

Theorem 4.1. *Let S_4 be the cubic spline approximation for a function $f \in C^4[a,b]$ on a set of equally spaced knots $t_0, t_1, \ldots, t_N$ with step size h, and assume $|f^{(4)}(t)| \le L$ for all $t \in [a,b]$. Then the approximation error satisfies:*

$$|f^{(k)}(t) - S_4^{(k)}(t)| \le c_k L h^{4-k}, \tag{4.1}$$

where $c_k \le 2$ ($k = 0, 1, 2, 3$) is a constant independent of the step size h.

Proof. For the proof, see [36].

Next, we prove a theorem that describes the approximation behavior of the fractional derivative of a function f.

Theorem 4.2. *Suppose $f(t) \in C^4[a,b]$ and $S_4(t)$ is the cubic spline approximation of ff on equally spaced knots$t_0, t_1, \ldots, t_N$ with step size h, and assume $|f^{(4)}(t)| \le L$ for all $t \in [a,b]$ and $1 < \alpha \le 2$. Then, there exists a constant C_2, independent of the partition length h, such that*

$$|{}_0^C D_t^\alpha f(t) - {}_0^C D_t^\alpha S_4(t)| \le C_2 h^{4-\alpha}. \tag{4.2}$$

Proof. Using the definition of the fractional derivative and Theorem 4.1, we have

$$\begin{aligned} |{}_0^C D_t^\alpha f(t) - {}_0^C D_t^\alpha S_4(t)| &\le \frac{1}{\Gamma(2-\alpha)} \int_0^t (t-\tau)^{1-\alpha} |f''(\tau) - S_4''(\tau)| d\tau, \\ &\le \frac{c_2 L h^2}{\Gamma(2-\alpha)} \int_0^t (t-\tau)^{1-\alpha} d\tau \\ &\le \frac{c_2 L h^2}{\Gamma(2-\alpha)} \frac{t^{2-\alpha}}{(2-\alpha)} \\ &= C_2 h^{4-\alpha}, \end{aligned} \tag{4.3}$$

where $C_2 = \dfrac{c_2 L \theta^{2-\alpha}}{\Gamma(3-\alpha)}$ and $t = \theta h$.

This theorem provides a bound on the error between the fractional derivative of the function $f(t)$ and its cubic spline approximation, demonstrating that the error decreases with step size h.

It is evident that our cubic B-spline approximation achieves an accuracy of $O(h^4)$, and the approximation error for the corresponding fractional derivative is $O(h^{4-\alpha})$.

Theorem 4.3. *Let $y_1 \le \cdots \le y_{n+m}$ be such that $y_i < y_{i+m}$ for $i = 1, 2, \ldots, n$. Let $N_1^m, \ldots, N_n^m$ be the associated normalized B-splines. Then there exists a dual set of linear functions $\lambda_1, \ldots, \lambda_n$ such that*

$$|\lambda_j f| \le (2m+1)9^{m-1} h_j^{-1/p} \|f\|_{L_p I_j}, \quad 1 \le p \le \infty, \tag{4.4}$$

where $I_j = (y_j, y_{j+m})$ and $h_j = y_{j+m} - y_j$ for $j = 1, 2, \ldots, n$.

Proof. We refer to [37] for the proof.

This theorem provides a bound on the norm of the dual linear functions associated with the B-splines, highlighting their effectiveness in approximating functions within the specified intervals.

Theorem 4.4. *Suppose $f \in L^p[0,1],\ 1 \le p \le \infty$ and*

$$Qf = \sum_{i=1}^{n} (\lambda_i f) N_i^m(x),$$

defines the B-spline approximation of order m, where $N_i^m,\ i = 1, \ldots, n$ is the associated B-spline basis and $\lambda_i,\ i = 1, \ldots, n$ is the corresponding dual basis. Then there exists a constant C (depending only on m and p) such that

$$\|f - Qf\|_p \le C d(f, S_m)_p, \tag{4.5}$$

where $S_m = span(N_1^m, \ldots, N_n^m)$.

Proof. We refer to [37] for the proof.

Theorems 4.1 and 4.2 address the convergence of B-spline approximation in the $C[0,1]$ space, while Theorems 4.3 and 4.4 explore the convergence properties of the B-spline approximation in the $L^p[0,1]$ space.

5. Numerical illustrations

This section provides numerical illustrations of the cubic B-spline method applied to various fractional-order Sturm–Liouville eigenvalue problems. All simulations are performed using Maple software with a precision of 40 digits.

For a fixed α, the experimental rate of convergence (C_{order}) for the kth eigenvalue is estimated using the formula:

$$C_{order} = \log_2 \frac{e(\lambda_k^{(M)})}{e(\lambda_k^{(M+1)})}, \tag{5.1}$$

where $e(\lambda_k^{(M)}) = |\lambda_k - \lambda_k^{(M)}|$ is the absolute error, with λ_k being the kth eigenvalue obtained with B-splines method.

In cases where the exact value of λ_k is not known, the results are compared with the most accurate value available from numerical methods. If the exact eigenvalue is not available, the experimental order of convergence can be calculated using the following alternative relationship:

$$C_{order} = \log_2 \left| \frac{\lambda_k^{(M)} - \lambda_k^{(M-1)}}{\lambda_k^{(M+1)} - \lambda_k^{(M)}} \right|. \tag{5.2}$$

Example 5.1. As a first example, consider the following fractional eigenvalue problem:

$${}_0^C D_t^{\alpha} y(t) + \lambda y(t) = 0, \qquad 0 \le t \le 1, \quad y(0) = y(1) = 0. \tag{5.3}$$

According to the boundary conditions (1.2), we set $a = c = 1$, $b = d = 0$, and $g(t) = -t$. When $\alpha = 2$, (5.3) is converted to the classical Sturm–Liouville problem, where the exact eigenvalues are $\lambda_k = (k\pi)^2$.

We have solved (5.3) for various values of M and α. The numerical results for this example are summarized in Tables 1 and 2. The first four eigenfunctions are illustrated in Fig. 1.

Remark 1. The bold line in the second column of Table 2 indicates that no eigenvalue was found for $\alpha = 1.6$. Additionally, only two eigenvalues were identified for $\alpha = 1.7$, whereas for larger values of α, a greater number of eigenvalues were obtained.

Remark 2. As shown in Table 1, the computational order of convergence increases with the number of basis functions, sometimes reaching up to order 9. However, in Table 5, the opposite trend is observed, where the order of convergence decreases as the number of basis functions increases. This fluctuation is attributed to rounding errors during calculations, which stem from the rapid increase in the condition number of the coefficient matrix, leading to an ill-conditioned problem. Consequently, it can be concluded that this method, similar to other spectral methods, is most effective for approximating eigenvalues with low indices.

Table 1
Eigenvalues for different M and $\alpha = 2$, Example 5.1.

	Exact value	Cubic B-spline method				Method of [38]
		$M = 4$	$M = 5$	$M = 6$	C_{order}	$N = 800$
λ_1	9.86960440108	9.86960618773	9.86960445697	9.86960440284	5.003	9.86960440
λ_2	39.4784176043	39.4785313533	39.4784211773	39.4784177161	4.993	39.47841760
λ_3	88.8264396098	88.8277231818	88.8264802302	88.8264408825	4.981	88.82643963
λ_4	157.913670417	157.920771429	157.913898028	157.913677564	4.963	157.91367042
λ_5	246.740110027	246.766677195	246.740975252	246.740137266	4.939	246.74011064
λ_6	355.305758439	355.383868167	355.308331336	355.305839698	4.922	355.30575850
λ_7	483.610615653	483.810870736	483.617075532	483.610820343	4.953	483.61061573
λ_8	631.654681670	632.151525770	631.669023169	631.655137240	5.119	631.65468191

Table 2
Eigenvalues for different α and $M = 6$, Example 5.1.

	$\alpha = 1.6$	$\alpha = 1.7$	$\alpha = 1.8$	$\alpha = 1.9$	$\alpha = 1.95$	$\alpha = 2$
λ_1	–	9.93290085202	9.45685689918	9.51414313433	9.66077192756	9.86960440284
λ_2		23.2509629280	28.4768793038	33.5956717800	36.4045246457	39.4784177161
λ_3			62.2003797920	73.0390189246	80.3290451049	88.8264408825
λ_4			97.0632438704	124.418538901	140.076242976	157.913677564
λ_5			155.450168859	191.146080458	216.597689946	246.740137266
λ_6			196.595983044	267.945277074	308.347659136	355.305839698
λ_7			301.527451554	361.086691842	416.740717999	483.610820343
λ_8				461.910582662	539.878336773	631.655137240

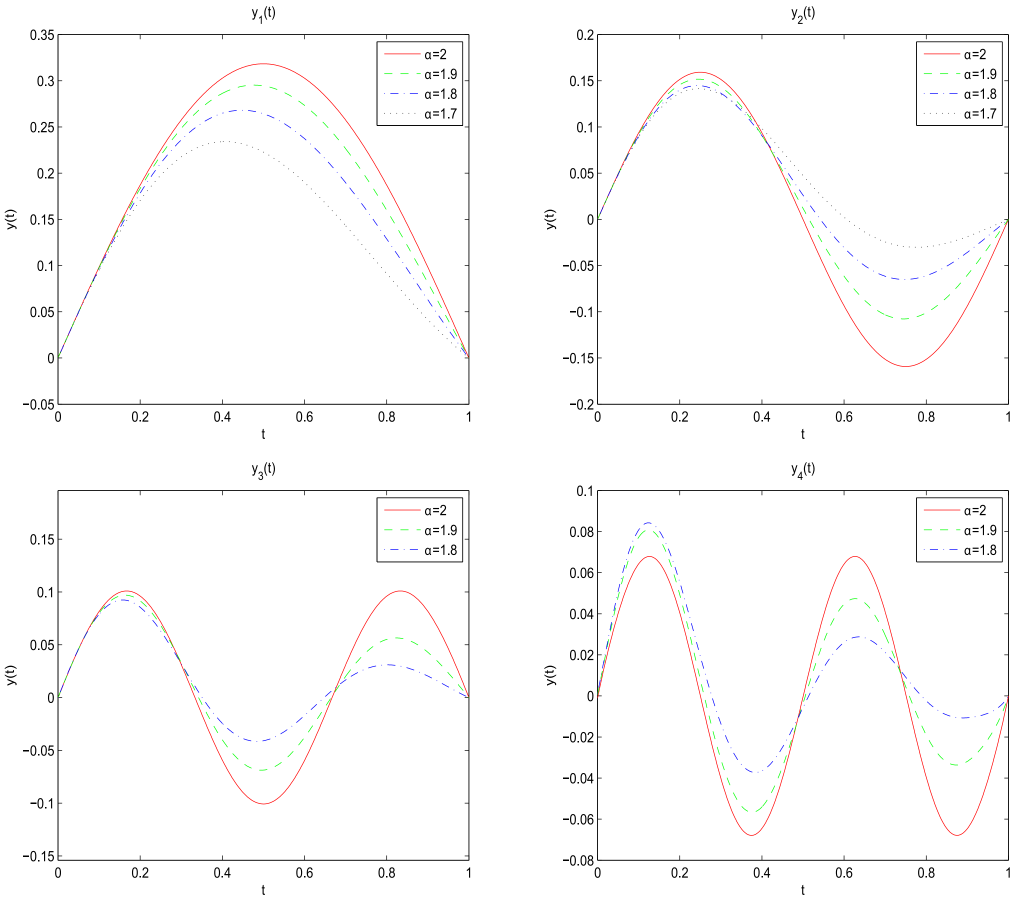

Fig. 1. First four eigenfunctions for Example 5.1.

Table 3
Eigenvalues for different M and $\alpha = 2$, Example 5.2.

	Exact value	Cubic B-spline method				Method of [38]
		$M = 4$	$M = 5$	$M = 6$	C_{order}	$N = 500$
λ_1	20.7922884552	20.7922899428	20.7922885400	20.7922884603	4.129	20.79228809
λ_2	82.4191538209	82.4192215567	82.4191563955	82.4191539382	4.727	82.41914815
λ_3	185.130596097	185.131328957	185.130619533	185.130596996	4.979	185.13056780
λ_4	328.926615284	328.930957494	328.926736162	328.926619480	5.177	328.92652741
λ_5	513.807211381	513.827812869	513.807659939	513.807225894	5.537	513.80700157
λ_6	739.772384388	739.867505601	739.773742584	739.772425399	6.153	739.77196118
λ_7	1006.82213431	1007.27904864	1006.82577381	1006.82223458	7.001	1006.82137649
λ_8	1314.95646113	1317.07167732	1314.96562495	1314.95681361	7.901	1314.95521911
λ_9	1664.17536487	1672.99150064	1664.19801725	1664.17581097	8.629	1664.17346838
λ_{10}	2054.47884552	2085.92400412	2054.53520493	2054.47969549	9.143	2054.47611287

Table 4
Eigenvalues for different α and $M = 6$, Example 5.2.

	$\alpha = 1.7$	$\alpha = 1.8$	$\alpha = 1.85$	$\alpha = 1.9$	$\alpha = 1.95$	$\alpha = 2$
λ_1	26.3237434500	21.7289811447	21.0420109762	20.7298857703	20.6677970633	20.7922884603
λ_2	40.3748707739	56.9308056705	63.3253037845	69.4887878652	75.7710030594	82.4191539382
λ_3		134.502660200	141.496580687	153.001682743	167.652674234	185.130596996
λ_4		193.872966751	226.249238191	257.628332780	291.242156378	328.926619480
λ_5		345.301242035	358.714929178	398.913258101	451.064973692	513.807225894
λ_6		379.250189053	474.803645402	554.923994617	641.015156750	739.772425399
λ_7			669.700857122	752.919635979	867.315104864	1006.82223458
λ_8			798.338625351	956.444962052	1122.29309425	1314.95681361

Table 5
Eigenvalues for different M and $\alpha = 2$, Example 5.3.

	Cubic B-spline method				Method of [38]
	$M = 4$	$M = 5$	$M = 6$	C_{order}	$N = 500$
λ_1	2.6212994353	2.6213001429	2.6213001835	4.110	2.62130018
λ_2	8.3495327960	8.3495216744	8.3495213999	5.343	8.34952147
λ_3	20.160576391	20.160288254	20.160277625	4.761	20.16027739
λ_4	37.781581453	37.779584686	37.779950188	–	37.77949944
λ_5	61.253511230	61.245881895	61.245509830	4.358	61.24549782
λ_6	90.587866399	90.569396559	90.568177324	3.921	
λ_7	125.77767748	125.75417162	125.75093445	2.860	
λ_8	166.77698972	166.80261795	166.79524403	–	
λ_9	213.49018436	213.71675528	213.70184420	–	
λ_{10}	265.99018513	266.49852601	266.47118053	–	

Example 5.2. For the second example, consider the following eigenvalue problem:

$$\begin{aligned} &{}_{0}^{C}D_x^{\alpha} y(t) + \frac{\lambda}{(1+t)^2} y(t) = 0, \qquad 0 \le t \le 1, \\ &\quad y(0) = 0, \;\; y(1) = 0. \end{aligned} \tag{5.4}$$

For $\alpha = 2$, the exact eigenvalues of (5.4) are given by:

$$\lambda_k = (k\pi/\ln 2)^2 + 1/4.$$

The corresponding eigenfunctions are:

$$y_k(t) = c_k \sqrt{1+t}\,\sin\left(\frac{k\pi \ln(1+t)}{\ln 2}\right),$$

where c_k is a constant coefficient [38]. The numerical results for this example are presented in Tables 3 and 4, and the eigenfunctions are illustrated in Fig. 2.

Example 5.3. Our next example is devoted to the following Sturm–Liouville problem:

$$\begin{aligned} &{}_{0}^{C}D_t^{\alpha} y(t) + (2\lambda e^t - 5\sin \pi t) y(t) = 0, \qquad 0 \le t \le 1, \\ &\quad y(0) = y'(0) \neq 0, \;\; y(1) = 0. \end{aligned} \tag{5.5}$$

In this example, the parameters are set as $a = 1$, $b = -1$, $c = 1$, $d = 0$, and the function $g(t)$ is assumed to be $-(t+1)/2$.

Tables 5 and 6 present the numerical results for this problem, comparing them with results from [38] for different values of α. To evaluate the eigenfunctions, we use the auxiliary condition $y'(1) = 1$, and the results are plotted in Fig. 3.

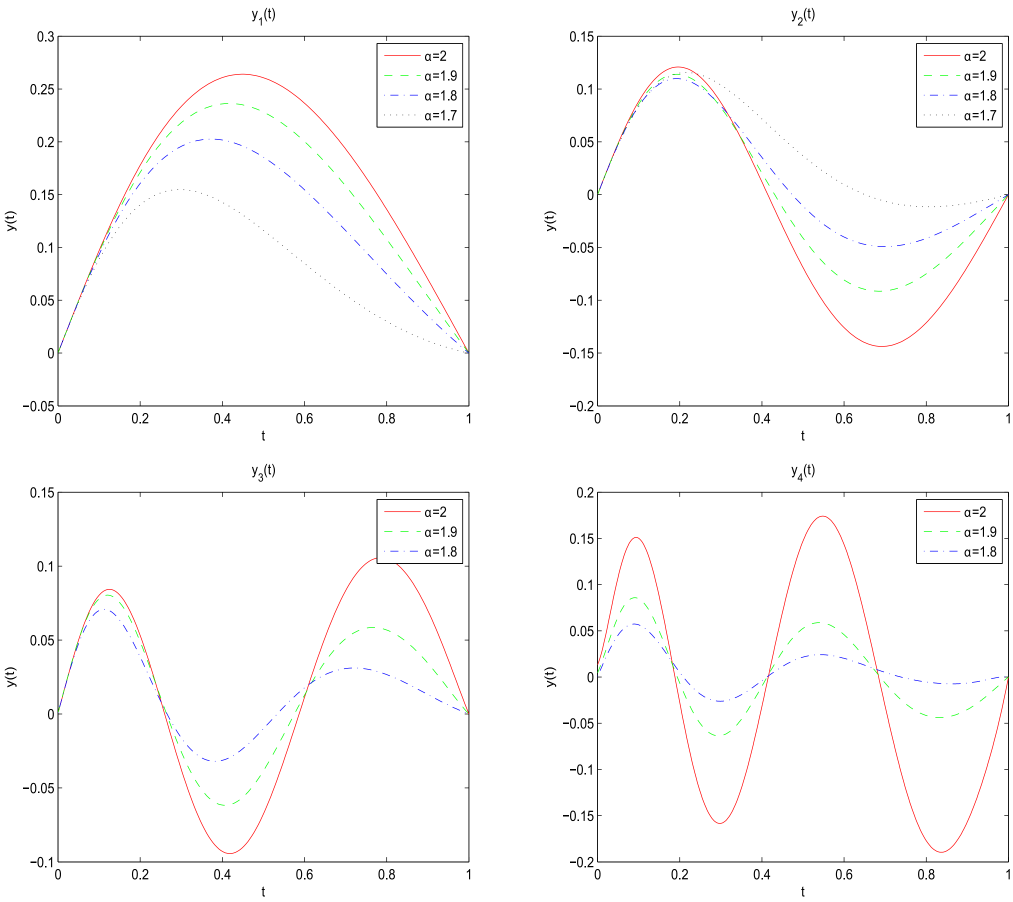

Fig. 2. First four eigenfunctions for Example 5.2.

Table 6
Eigenvalues for different α and $M = 6$, Example 5.3.

	Cubic B-spline method, $M = 6$				Method of [38] $N = 800$	
	$\alpha = 1.8$	$\alpha = 1.85$	$\alpha = 1.9$	$\alpha = 1.95$	$\alpha = 1.85$	$\alpha = 1.95$
λ_1	2.4809710061	2.5083125020	2.5408887681	2.5785603317	2.50831250	2.57856033
λ_2	6.4250041157	6.8263670743	7.2770439770	7.7826673786	6.82636707	7.78266742
λ_3	13.934784148	15.191208145	16.631517814	18.278662699	15.19120752	18.27866231
λ_4	24.257866560	26.945889357	30.049142312	33.634633407	26.94588552	33.63463032
λ_5	37.310174253	41.977289139	47.432832215	53.803950345	41.97727381	53.80393654
λ_6	52.853190284	60.132007716	68.678813381	78.731592795		
λ_7	70.935536388	81.358730635	93.721466356	108.37384187		
λ_8	91.265973382	105.53129255	122.48780427	142.69083130		

Example 5.4. Consider the following Sturm–Liouville problem [39]

$$\begin{aligned} &{}_0^C D_x^\alpha y(x) + (\lambda - \frac{1}{(x+0.1)^2})y(x) = 0, \qquad 0 \le x \le \pi, \\ &\qquad y(0) = 0, \;\; y(\pi) = 0. \end{aligned} \tag{5.6}$$

By placing $x = \pi t$, the problem (5.8) can be recast as:

$$\begin{aligned} &{}_0^C D_t^\alpha u(t) + (\overline{\lambda} - \frac{1}{(t+0.1/\pi)^2})u(t) = 0, \qquad 0 \le t \le 1, \\ &\qquad u(0) = 0, \;\; u(1) = 0, \end{aligned} \tag{5.7}$$

where $u(t) = y(\pi t)$ and $\overline{\lambda} = \pi^2 \lambda$.

Tables 7 and 8 present the first eigenvalues and compare them with results obtained using the differential quadrature (DQ) method for different values of aa. The graph of some of the first eigenfunctions is shown in Fig. 4.

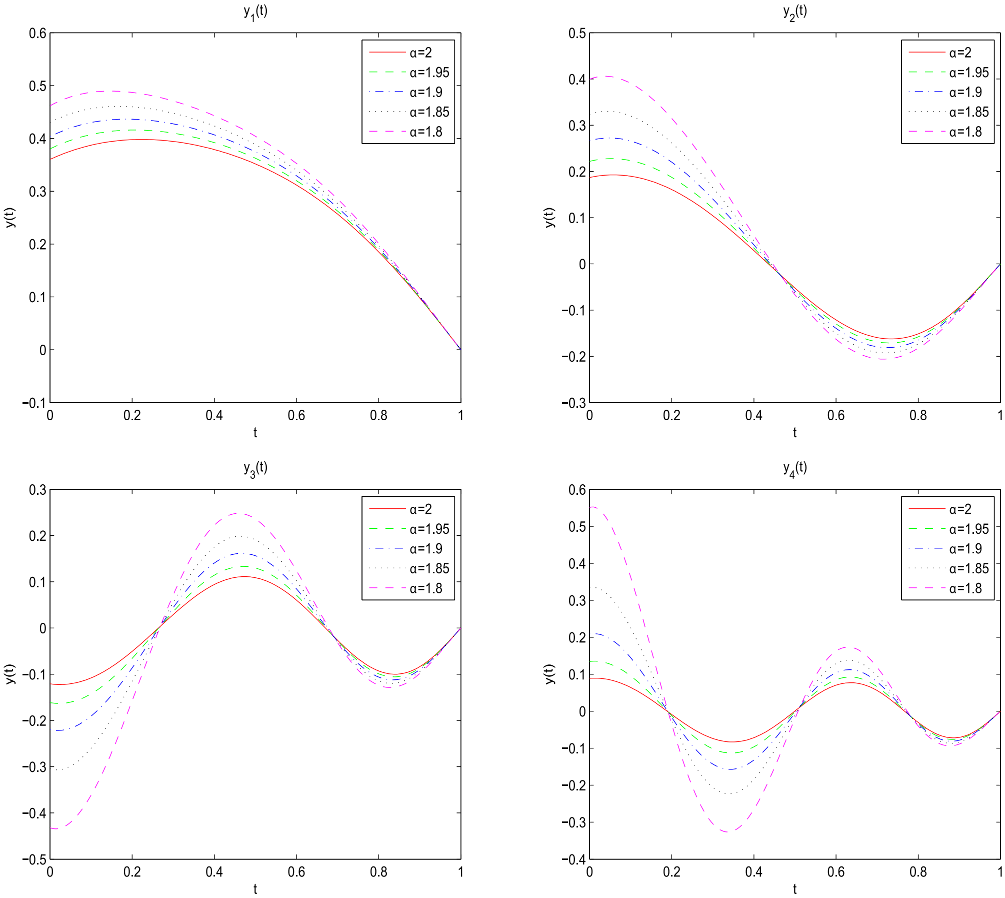

Fig. 3. First four eigenfunctions for Example 5.3.

Table 7
Eigenvalues for $\alpha = 2$, Example 5.4.

	Cubic B-spline method				DQ method [40]
	$M = 4$	$M = 5$	$M = 6$	$M = 7$	$N = 50$
λ_1	1.5198628252	1.5198657617	1.5198658209	1.5198658211	1.5198658
λ_2	4.9432763314	4.9433099362	4.9433098380	4.9433098227	4.9433098
λ_3	10.284657321	10.284666733	10.284662818	10.284662651	10.284663
λ_4	17.560509211	17.559982884	17.559958650	17.559957775	17.559958
λ_5	26.786244308	26.782959726	26.782866431	26.782863262	26.782863
λ_6	37.978314416	37.964711898	37.964435272	37.964426171	37.964426
λ_7	51.155753626	51.114071968	51.113380841	51.113358483	51.113358
λ_8	66.313065726	66.238026838	66.236498085	66.236449287	66.236448
λ_9	83.462087888	83.342152907	83.339062897	83.338965533	83.338662
λ_{10}	102.81580548	102.43101356	102.42517520	102.42499427	102.42499

Example 5.5. The last example solves the fractional Sturm–Liouville problem [38]

$$\begin{aligned} &{}_0^C D_t^{\alpha} y(t) + (\lambda + 10\sin \pi t) y(t) = 0, \qquad 0 \le t \le 1, \\ &y(0) = 0, \;\; y(1) = 0. \end{aligned} \tag{5.8}$$

In Tables 9 and 10 we compared the results for some first eigenvalues with Lagrange polynomial integration (LPI) method [38]. The illustrative graphs are plotted in Fig. 5.

6. Conclusion

In the present paper, we addressed a class of fractional order Sturm–Liouville problems using the cubic B-spline approximation method. This approach involved expanding both the unknown solution and the variable coefficients in terms of cubic B-spline

Table 8
Eigenvalues for different α and $M = 6$, Example 5.4.

	Cubic B-spline method, $M = 6$					
	$\alpha = 1.7$	$\alpha = 1.8$	$\alpha = 1.85$	$\alpha = 1.9$	$\alpha = 1.95$	$\alpha = 2$
λ_1	1.1762602431	1.2762493377	1.3313060888	1.3900951543	1.4528678849	1.5198658209
λ_2	3.1712104242	3.6563270302	3.9344046866	4.2395154239	4.5746837854	4.9433098380
λ_3	6.1804410977	7.1426799126	7.7742198536	8.5035301572	9.3369797826	10.284662818
λ_4	8.8279735119	11.236085752	12.552837531	14.017404009	15.672858148	17.559958650
λ_5		16.812440475	18.665021812	20.941331691	23.633884888	26.782866431
λ_6		21.997982608	25.333295729	28.969604888	33.126920367	37.964435272
λ_7		30.430905672	33.773630244	38.489969966	44.249046170	51.113380841
λ_8		35.057242820	41.959298312	48.920833043	56.866517508	66.236498085
λ_8			53.089405782	61.029723280	71.125866396	83.339062897
λ_8			62.068586948	73.743654041	86.840949967	102.42517520

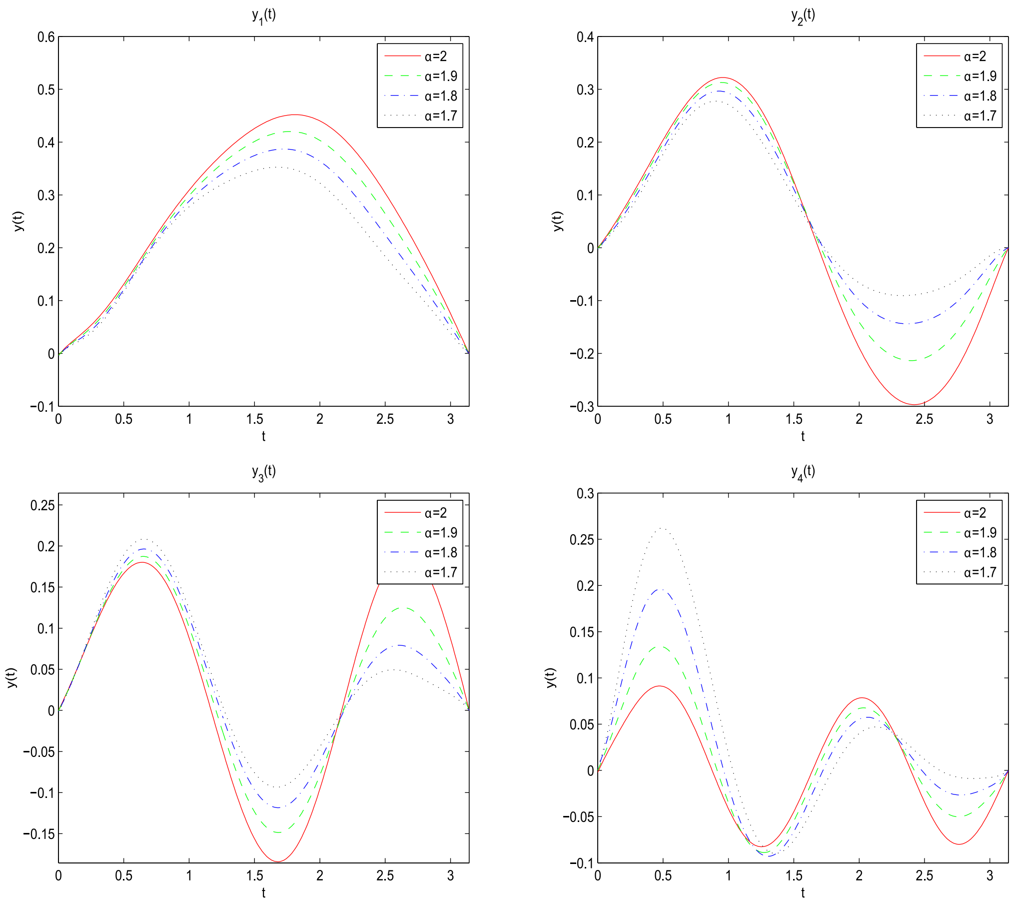

Fig. 4. First four eigenfunctions for Example 5.4.

polynomials. By substituting these expansions into the differential equation, the problem was transformed into a system of algebraic equations involving the eigenvalue parameter λ, which is essential for determining the eigenvalues of the Sturm–Liouville problem.

We demonstrated that the existence of a non-trivial solution to the Sturm–Liouville problem is equivalent to solving a homogeneous system of linear equations, which led to the formulation of an extended eigenvalue problem. Solving this eigenvalue problem provided approximations for the eigenvalues of the original problem. These approximations showed high accuracy, particularly for lower eigenvalue indices, although the accuracy tended to decrease as the index increased.

Our results indicate that the cubic B-spline approximation method offers a competitive level of accuracy when compared with other numerical methods for solving fractional order Sturm–Liouville problems. The methods ability to handle complex boundary conditions and fractional derivatives effectively positions it as a valuable tool in the numerical analysis of such problems.

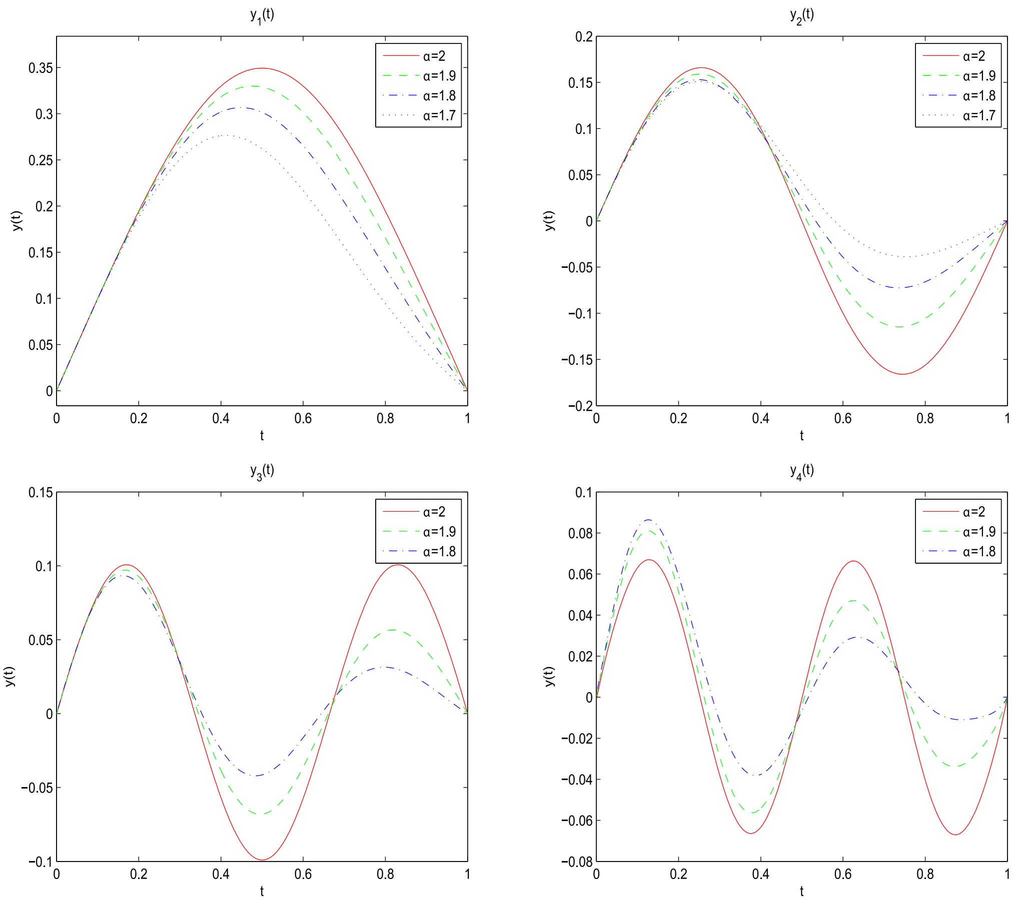

Fig. 5. First four eigenfunctions for Example 5.5.

Table 9
Eigenvalues for $\alpha = 2$, Example 5.5.

	Cubic B-spline method				Method of [38]
	$M = 4$	$M = 5$	$M = 6$	$M = 7$	$N = 800$
λ_1	1.3455560055	1.3455507896	1.3455505124	1.3455504958	1.34555049
λ_2	32.655942281	32.655832461	32.655828821	32.655828677	32.65582867
λ_3	82.288915091	82.287718525	82.287681563	82.287680352	82.28768033
λ_4	151.46319383	151.45647099	151.45626169	151.45625507	151.45625747
λ_5	240.34348374	240.31798986	240.31718279	240.31715740	240.31715718
λ_6	348.97903300	348.90324013	348.90081542	348.90073907	348.90073667
λ_7	477.41894685	477.22248023	477.21634927	477.21615570	477.21614970
λ_8	625.77453655	625.28109301	625.26741291	625.26697998	625.26696665
λ_9	794.42033607	793.08353424	793.05573236	793.05485253	
λ_{10}	984.71501412	980.63494879	980.58227596	980.58061764	

Table 10
Eigenvalues for different α and $M = 6$, Example 5.5.

	Cubic B-spline method, $M = 6$				Method of [38], $N = 800$	
	$\alpha = 1.7$	$\alpha = 1.8$	$\alpha = 1.85$	$\alpha = 1.9$	$\alpha = 1.85$	$\alpha = 1.9$
λ_1	0.9864233068	0.7283148778	0.7766494049	0.9036756920	0.77664936	0.90367565
λ_2	16.258734855	21.506911148	24.052043483	26.704709058	24.05204303	26.70470868
λ_3		55.714493865	60.424832501	66.504282245	60.42482813	66.50427914
λ_4		90.508181965	103.85896311	117.94233229	103.85889843	117.94227903
λ_5		149.14711327	163.92259999	184.73221518	163.92253966	184.73216670

(*continued on next page*)

Table 10 (*continued*).

	Cubic B-spline method, $M = 6$				Method of [38], $N = 800$	
	$\alpha = 1.7$	$\alpha = 1.8$	$\alpha = 1.85$	$\alpha = 1.9$	$\alpha = 1.85$	$\alpha = 1.9$
λ_6		190.05030048	225.57019981	261.52847268	225.57013005	261.52839381
λ_7			310.57465734	354.70151475	310.57444773	354.70131023
λ_8			385.73850552	455.51227342	385.73816449	455.51185987
λ_8			500.50189191	574.88708572		
λ_8			580.50189191	698.29687858		

CRediT authorship contribution statement

Arezu Aghazadeh: Writing – original draft, Software, Conceptualization. **Mehrdad Lakestani:** Writing – review & editing, Supervision, Resources, Methodology, Conceptualization.

Acknowledgments

The authors would like to thank the expert referees for their carefully reading and useful suggestions and comments which led to improvement the paper. This paper is supported by the Research Affairs Office of the University of Tabriz, Tabriz, Iran under Grant No. S-3044, as part of a postdoctoral research project.

References

[1] M.A. Al-Gwaiz, Sturm-Liouville Theory and Its Applications, Springer, London, 2008.
[2] Q.M. Al-Mdallal, M.I. Syam, The Chebyshev collocation-path following method for solving sixth-order Sturm-Liouville problems, Appl. Math. Comput. 232 (2014) 391–398.
[3] M. Syam, M. Elgamel, H. Siyyam, An effcient technique for fnding the eigenvalues of fourth-order Sturm-Liouville problems, Chaos Solitons Fractals 39 (2) (2009) 659–665.
[4] M.M. Djrbashian, A boundary value problem for a Sturm–Liouville type differential operator of fractional order, Izv. Acad. Nauk. Armjan. SSR Ser. Mat. 5 (2) (1970) 71–96.
[5] A.M. Nahusev, A Sturm–Liouville problem for a second order ordinary differential equation with fractional derivatives in the lower terms, Dokl. Akad. Nauk SSSR 234 (2) (1977) 308–311.
[6] M. Sattari, M. Arabameri, Compact finite difference scheme for numerical solution of caputo-fabrizio fractional riccati differential equations, Comput. Methods Differ. Equations (2025) http://dx.doi.org/10.22034/cmde.2025.60832.2602, accepted.
[7] Q. Al-Mdallal, An effcient method for solving fractional Sturm-Liouville problems, Chaos Solitons Fractals 40 (2009) 183–189.
[8] Q.M. Al-Mdallal, On the numerical solution of fractional Sturm-Liouville problems, Int. J. Comput. Math. 87 (12) (2010) 2837–2845.
[9] S. Abbasbandy, A. Shirzadi, Homotopy analysis method for multiple solutions of the fractional Sturm–Liouville problems, Numer. Algorithms 54 (2010) 521–532.
[10] A. Neamaty, R. Darzi, S. Zaree, Haar wavelet operational matrix of fractional order integration and its application for eigenvalues of fractional Sturm–Liouville problem, World Appl. Sci. J. 15 (2011) 1780–1785.
[11] Z. Shi, Y. Cao, Application of haar wavelet method to eigenvalue problems of high order differential equations, Appl. Math. Model. 36 (2012) 4020–4026.
[12] Y. Luchko, Initial–boundary-value problems for the one-dimensional time-fractional diffusion equation, Fract. Calc. Appl. Anal. 15 (1) (2012) 141–160, http://dx.doi.org/10.2478/s13540-012-0010-7.
[13] M. Klimek, T. Odzijewicz, A.B. Malinowska, Variational methods for the fractional Sturm-Liouville problem, J. Math. Anal. Appl. 416 (1) (2014) 402–426.
[14] P. Antunes, R. Ferreira, An augmented-RBF method for solving fractional Sturm–Liouville eigenvalue problems, SIAM J. Sci. Comput. 37 (1) (2015) 515–535.
[15] B. Jin, R. Lazarov, X. Lu, Z. Zhou, A simple fnite element method for boundary value problems with a Riemann-Liouville derivative, J. Comput. Appl. Math. 293 (2015).
[16] Q. Al-Mdallal, M. Al-Refai, M. Syam, M. Sirhin, Theoretical and computational perspectives on the eigenvalues of fourth order fractional Sturm–Liouville problem, Int. J. Comput. Math. (2017).
[17] B. Kashkari, M. Syam, An accurate method for solving a class of fractional Sturm–Liouville eigenvalue problems, Results Phys. (2018).
[18] Y. Khalili, M. Dehghan, Computing high-index eigenvalues for the Sturm–Liouville equation with Robin boundary conditions, Comput. Methods Differ. Equations (2024) http://dx.doi.org/10.22034/cmde.2024.60228.2566, accepted.
[19] M.H. Derakhshan, A. Ansari, Numerical approximation to prabhakar fractional Sturm-Liouville problem, Comput. Appl. Math. (2019) http://dx.doi.org/10.1007/s40314-019-0826-4.
[20] A. Aghazadeh, Y. Mahmoudi, F.D. Saei, Legendre approximation method for computing eigenvalues of fourth order fractional Sturm-Liouville problem, Math. Comput. Simulation 206 (2023) 286–301.
[21] A. Aghazadeh, Y. Mahmoudi, On approximating eigenvalues and eigenfunctions of fractional order Sturm–Liouville problems, Comput. Methods Differ. Equations 11 (4) (2023) 811–821.
[22] M. Yaseen, M. Abbas, B. Ahmad, Numerical simulation of the nonlinear generalized time-fractional Klein-Gordon equation using cubic trigonometric B-spline functions, Math. Methods Appl. Sci. 44 (1) (2021) 901–916.
[23] M. Abbas, M.K. Iqbal, B. Zafar, S.B.M. Zin, New cubic b-spline approximations for solving non-linear third-order korteweg-de vries equation, Indian J. Sci. Technol. 12 (6) (2019) 1–9.
[24] M. Shafiq, M. Abbas, K.M. Abualnaja, M.J. Huntul, A. Majeed, T. Nazir, An effcient technique based on cubic B-spline functions for solving time-fractional advection diffusion equation involving Atangana-Baleanu derivative, Eng. Comput. 38 (2022) 901–917.
[25] T. Akram, M. Abbas, A.I. Ismail, N.Hj.M. Ali, D. Baleanu, Extended cubic B-splines in the numerical solution of time fractional telegraph equation, Adv. Differ. Equ. (1) (2019) http://dx.doi.org/10.1186/s13662-019-2296-9.
[26] I. Ali, M. Yaseen, I. Akram, Utilizing cubic B-spline collocation technique for solving linear and nonlinear fractional integro-differential equations of Volterra and Fredholm types, Fractal Fract. 8 (5) (2024) 268, http://dx.doi.org/10.3390/fractalfract8050268.

[27] Y. Edrisi-Tabriz, M. Lakestani, M. Razzaghi, Study of B-spline collocation method for solving fractional optimal control problems, Trans. Inst. Meas. Control (2021) 1–13, http://dx.doi.org/10.1177/0142331220987537.
[28] T. Akram, M. Abbas, A. Ali, A numerical study on time fractional Fisher equation using an extended cubic B-spline approximation, J. Math, Comput. Sci. 22 (1) (2021) 85–96.
[29] M.K. Iqbal, M. Abbas, T. Nazir, N. Ali, Application of new quintic polynomial B-spline approximation for numerical investigation of Kuramoto-Sivashinsky equation, Adv. Differ. Equ. 558 (2020).
[30] N. Khalid, M. Abbas, M.K. Iqbal, J. Singh, A.I.M. Ismail, A computational approach for solving time fractional differential equation via spline functions, Alex. Eng. J. 59 (5) (2020) 3061–3078.
[31] M. Yaseen, M. Abbas, T. Nazir, D. Baleanu, A finite difference scheme based on cubic trigonometric B-splines for time fractional diffusion-wave equation, Adv. Differ. Equ. 274 (2017) 29.
[32] S.T. Mohyud-Din, T. Akram, M. Abbas, A.I. Ismail, N.H. Ali, A fully implicit finite difference scheme based on extended cubic B-splines for time fractional advection-diffusion equation, Adv. Differ. Equ. (2018) 109.
[33] W. Gautschi, The condition of polynomials in power form, Math. Comp. 33 (145) (1979) 343–352.
[34] J.C. Goswami, A.K. Chan, Fundamentals of Wavelets: Theory, Algorithms, and Applications, John Wiley and Sons Inc, 1999, http://dx.doi.org/10.1002/9780470926994.
[35] C. De Boor, A Practical Guide To Spline, Springer-Verlag, New York, 1978.
[36] J. Stoer, R. Bulirisch, Introduction To Numerical Analysis, Springer-Verlage, New York, 1992.
[37] L. Schumaker, Spline Functions: Basic Theory, Cambridge University Press, Nashville, London, 2007.
[38] M.K. Sadabad, A.J. Akbarfam, An efficient numerical method for estimating eigenvalues and eigenfunctions of fractional sturm-liouville problems, Math. Comput. Simulation 185 (2021) 547–569.
[39] H. Farzana, S. Kumar Bhowmik, M.A. Alim, Bernstein collocation technique for a class of Sturm–Liouville problems, Heliyon 10 (2024) e28888.
[40] U. Yücel, Approximations of Sturm-Liouville eigenvalues using differential quadrature (DQ) method, J. Comput. Appl. Math. 192 (2006) 310–319.