TWMS J. App. and Eng. Math. V.16, N.8, 2026, pp. 1018-1034

NUMERICAL APPROXIMATION OF ABC-TYPE FRACTIONAL EQUATIONS USING A VIETA-LUCAS POLYNOMIAL

GHADAH S.E. NOMAN[1,2,*], D.D. PAWAR[2], §

Abstract. This paper presents a spectral collocation method based on shifted Vieta-Lucas polynomials (SVLPs) for the numerical approximation of multi-term variable-order fractional differential equations (MT-VOFDEs) involving the Atangana-Baleanu-Caputo (ABC) fractional derivative in variable-order form. A novel operational matrix of the ABC derivative is derived in the SVLP framework, enabling an efficient transformation of the fractional system into an algebraic system.

The stability and convergence of the method are theoretically analyzed. The proposed SVLP-based approach is then applied to several test problems, including systems of MT-VOFDEs with known exact polynomial solutions. Numerical results demonstrate exponential convergence, high accuracy, and reduced computational cost compared to other polynomial-based collocation methods.

The orthogonality and recursive structure of the SVLPs contribute to computational efficiency and robustness. These findings highlight the effectiveness of the proposed method for solving complex fractional models governed by non-singular kernel operators.

Keywords: Atangana-Baleanu–Caputo derivative, Variable-order fractional differential equations, spectral collocation method, shifted Vieta-Lucas polynomials, operational matrix.

AMS Subject Classification: 26A33, 42C05, 65L60.

1. Introduction

Fractional differential equations (FDEs) have gained increasing attention in recent decades due to their ability to accurately capture memory effects, hereditary behavior, and nonlocal dynamics inherent in numerous real-world systems. Compared to classical integer-order models, FDEs offer enhanced modeling accuracy in applications ranging from viscoelastic materials and anomalous diffusion to control systems, signal processing, and population dynamics [1, 2, 3, 4, 5, 6].

[1] Department of Mathematics, Faculty of Applied Sciences, Taiz University, Yemen.
e-mail: ghadah.sultan.esmail@gmail.com; ORCID:https://orcid.org/0009-0002-2763-7637.
[2] School of Mathematical Sciences, Swami Ramanand Teerth Marathwada University, Nanded-431606, India.
e-mail: dypawar@yahoo.com; ORCID: https://orcid.org/0000-0001-8986-5243.
[*] Corresponding author
§ Manuscript received: May 23, 2025; accepted: September 24, 2025.
TWMS Journal of Applied and Engineering Mathematics, Vol.16, No.8;

Fractional calculus provides a natural extension of classical differentiation by incorporating non-integer orders of differentiation, allowing systems with memory and hereditary properties to be described more realistically. This nonlocality is especially useful in fields like bioengineering, fluid mechanics, and finance, where past states influence present behavior. Unlike traditional integer-order derivatives, fractional derivatives enable the modeling of processes with fading memory and long-range dependence, which are frequently observed in natural and engineered systems. While traditional definitions, such as the Riemann-Liouville, Caputo, and Grünwald-Letnikov operators-have been widely used, their singular kernel structures often lead to computational inefficiencies and limitations in physical interpretability. To overcome these drawbacks, modern operators with non-singular kernels, such as the Caputo-Fabrizio and Atangana-Baleanu derivatives, have been proposed [7, 8, 9, 10, 11]. Among these, the ABC derivative has gained prominence for its use of the Mittag-Leffler kernel, which offers smoother and more physically meaningful representations of memory effects [12, 13]. The ABC operator facilitates crossover behaviors, such as Gaussian to non-Gaussian and stretched exponential to power-law transitions-making it particularly suitable for modeling anomalous diffusion, biological dynamics, and thermal transport processes [14, 15, 16, 17].

In parallel, the development of variable-order fractional calculus (VOFC) has introduced a dynamic dimension to modeling. First introduced by Samko and Ross [18], VOFC allows the order of differentiation to vary with time, space, or other system parameters, enabling adaptive modeling of systems with evolving memory. This flexibility is especially important in simulating processes like adaptive control, memory-based diffusion, and spatially heterogeneous media. However, the mathematical and numerical complexities of solving variable-order models, especially in multi-term forms-remain a significant challenge.

To address these challenges, numerous numerical strategies have been introduced. Li and Wu [19] proposed reproducing kernel methods; Sayed and Baleanu [20] developed a Jacobi-based spectral approach, Tural and Dincel [21] applied Bernstein polynomials, and Amin et al. [22] presented a space-time spectral collocation method utilizing shifted Legendre and Chebyshev polynomials for solving variable-order fractional Fokker–Planck equations. Recently, significant attention has been directed toward numerical schemes capable of handling variable-order ABC fractional derivatives in complex multi-dimensional settings. For instance, Sharma and Rajeev [23] conducted a comprehensive numerical study of a three-dimensional fractional advection-diffusion equation of variable order involving a modified Atangana-Baleanu-Caputo derivative. Although these methods provide accurate results, their performance can suffer when dealing with highly variable orders or requiring flexible node distributions.

In addition, recent works in the literature have addressed related problems using different fractional operators and theoretical techniques. For instance, Y. Awad [24] investigated nonlinear hybrid fractional differential equations in the setting of Banach algebras, while Nikam and Manjarekar [25] studied applications of variable-order Riemann–Liouville fractional calculus in complex systems. Rahou et al. [26] analyzed the existence of solutions to implicit fractional differential equations involving the Riesz–Caputo derivative. Furthermore, Karaoglan and Erdal [27] presented a foundational spectral framework for fractional systems governed by generalized M-type derivatives. These studies reflect the growing interest in both theoretical and numerical treatments of generalized fractional operators and emphasize the need for robust numerical techniques tailored to variable-order fractional models.

The need remains for a computational framework that provides both spectral accuracy and flexibility in handling variable-order kernels, particularly those governed by non-singular memory operators like ABC. In this work, we address this gap by developing a novel spectral collocation method using SVLPs. These polynomials form an orthogonal basis with advantageous recurrence relations and convergence behavior, making them ideal for high-precision numerical computation. The proposed method leverages SVLPs to construct an operational matrix for the ABC fractional derivative with variable order, allowing the transformation of complex MT-VOFDEs into solvable algebraic systems.

The contributions of this paper are threefold:

- We construct, for the first time, an operational matrix of the variable-order ABC derivative using shifted Vieta-Lucas polynomials.
- We develop a stable and efficient spectral collocation method for VO-FDEs, applicable to both scalar equations and coupled systems.
- We validate the method through extensive numerical experiments, reporting error norms (L_2 and L_∞), convergence rates, and CPU times. Comparisons with Chebyshev-based and Bernstein-based collocation schemes demonstrate the superior accuracy and efficiency of the proposed SVLP approach.

The remainder of this paper is organized as follows. Section 2 introduces the necessary preliminaries, including definitions of fractional operators, the ABC derivative, and shifted Vieta-Lucas polynomials. Section 3 formulates the operational matrix associated with the VO-ABC derivative. Section 4 presents the proposed numerical scheme based on this framework. Section 5 discusses the error analysis to establish the reliability and convergence of the method. Section 6 provides numerical examples for both scalar and coupled VO-FDEs. Finally, Section 7 concludes the paper with remarks on the method's significance and possible future research directions.

We consider MT-VOFDEs of the form:

$$\sum_{i=1}^{\mathfrak{n}} \Upsilon_i(\mathfrak{t})\, {}^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}\mathrm{U}(\mathfrak{t}) = \mathcal{Z}(\mathfrak{t}, \mathrm{U}(\mathfrak{t})), \quad 0 \leq \mathfrak{t} \leq 1, \tag{1}$$

$$\mathrm{U}(0) = \mathrm{U}_0, \tag{2}$$

where $\mathfrak{n}$ is a positive integer, $\xi_i(\mathfrak{t}) \in [0,1]$ for $i = 1, 2, \ldots, \mathfrak{n}$ are variable-order functions, and $\Upsilon_i(\mathfrak{t})$ are known coefficient functions. The function $\mathrm{U}(\mathfrak{t})$ is assumed to be continuously differentiable. The operator ${}^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}$ denotes the variable-order Atangana–Baleanu–Caputo derivative defined in Definition 2.1.

2. Preliminaries

In this section, we recall some basic concepts and definitions that will be used throughout the paper, including variable-order fractional derivatives, the ABC operator, and SVLPs.

Definition 2.1. [8, 28] *The Atangana-Baleanu-Caputo fractional derivative* ${}^{ABC}\mathcal{D}^{\xi}$ *of order* $0 < \xi < 1$ *of the function* $\mathrm{U}(\mathfrak{t}) \in \mathcal{H}^1(a,b), a \in (-\infty, \mathfrak{t})$ *is formulated as follows:*

$${}^{ABC}\mathcal{D}^{\xi}\mathrm{U}(\mathfrak{t}) = \frac{N(\xi)}{1-\xi}\int_a^{\mathfrak{t}} E_\xi[-C(\xi)(\mathfrak{t}-s)^\xi]\mathrm{U}'(s)ds. \tag{3}$$

Where E_ξ *is the Mittag-Leffler function, defined as*

$$E_\xi(\mathfrak{t}) = \sum_{k=0}^{\infty} \frac{\mathfrak{t}^k}{\Gamma(k\xi+1)},$$

and

$$E_{\xi_1,\xi_2}(\mathfrak{t}) = \sum_{k=0}^{\infty} \frac{\mathfrak{t}^k}{\Gamma(k\xi_1+\xi_2)}.$$

Moreover, $N(\xi)$ is a normalization function such that $N(0) = N(1) = 1$ and $N(\xi) = 1-\xi+\frac{\xi}{\Gamma(\xi)}$, and $C(\xi) = \frac{\xi}{1-\xi}$.

For the variable-order derivative, Eq. (3) *can be rewritten as* [8]*:*

$$^{ABC}\mathcal{D}^{\xi(\mathfrak{t})}\mathrm{U}(\mathfrak{t}) = \frac{N(\xi(\mathfrak{t}))}{1-\xi(\mathfrak{t})}\int_a^{\mathfrak{t}} E_{\xi(\mathfrak{t})}[-C(\xi(\mathfrak{t}))(\mathfrak{t}-s)^{\xi(\mathfrak{t})}]\mathrm{U}'(s)ds. \tag{4}$$

It follows that

$$^{ABC}\mathcal{D}^{\xi(\mathfrak{t})}\mathfrak{t}^r = \begin{cases} 0 & \textit{for } r=0, \\ \frac{N(\xi(\mathfrak{t}))\mathfrak{t}^r\Gamma(r+1)}{1-\xi(\mathfrak{t})}E_{\xi(\mathfrak{t}),r+1}\left(\frac{-\xi(\mathfrak{t})}{1-\xi(\mathfrak{t})}\mathfrak{t}^{\xi(\mathfrak{t})}\right) & \textit{for } r=1,2,\dots \end{cases} \tag{5}$$

2.1. **Vieta-Lucas and Shifted Vieta-Lucas Polynomials.** We consider the VL polynomials, denoted by $\mathcal{VL}_{\mathfrak{m}}(\mathfrak{t})$, which are of degree $\mathfrak{m} \in N_0$, defined on the interval $[-2,2]$, and expressed in the following form [29]:

$$\mathcal{VL}_{\mathfrak{m}}(\mathfrak{t}) = 2\cos(\mathfrak{m}\theta), \quad \theta = \arccos\left(\frac{\mathfrak{t}}{2}\right), \quad \theta \in [0,\pi].$$

A recurrence formula can be used to define them:

$$\mathcal{VL}_{\mathfrak{m}}(\mathfrak{t}) = \mathfrak{t}\mathcal{VL}_{\mathfrak{m}-1}(\mathfrak{t}) - \mathcal{VL}_{\mathfrak{m}-2}(\mathfrak{t}), \quad \mathfrak{m} \geq 2,$$

$$\mathcal{VL}_0(\mathfrak{t}) = 2, \quad \mathcal{VL}_1(\mathfrak{t}) = \mathfrak{t}.$$

Furthermore, the VL polynomials can also be expressed as:

$$\mathcal{VL}_{\mathfrak{m}}(\mathfrak{t}) = \begin{cases} 2, & \mathfrak{m}=0, \\ \sum_{k=0}^{\lceil \frac{\mathfrak{m}}{2} \rceil}(-1)^k \frac{\mathfrak{m}(\mathfrak{m}-k-1)!}{k!(\mathfrak{m}-2k)!}\mathfrak{t}^{\mathfrak{m}-2k}, & \mathfrak{m} \geq 1, \end{cases} \tag{6}$$

where $\lceil \frac{\mathfrak{m}}{2} \rceil$ denotes the ceiling function.

These polynomials possess the property of orthogonality on the interval [-2,2] with respect to the weight function $\mathcal{W}(\mathfrak{t}) = \frac{1}{\sqrt{4-\mathfrak{t}^2}}$.

$$\langle \mathcal{VL}_{\mathfrak{m}}(\mathfrak{t}), \mathcal{VL}_{\imath}(\mathfrak{t})\rangle = \int_{-2}^{2} \mathcal{W}(\mathfrak{t})\mathcal{VL}_{\mathfrak{m}}(\mathfrak{t})\mathcal{VL}_{\imath}(\mathfrak{t})\mathrm{d}\mathfrak{t} = \begin{cases} 4\pi & \mathfrak{m}=\imath=0, \\ 2\pi & \mathfrak{m}=\imath\neq 0, \\ 0 & \mathfrak{m}\neq\imath. \end{cases}$$

As a new class of orthogonal polynomials, the shifted VL polynomials $\mathcal{VL}^*_{\mathfrak{m}}(\mathfrak{t})$ of degree $\mathfrak{m}$, defined on the closed interval [0,1] as

$$\mathcal{VL}^*_{\mathfrak{m}}(\mathfrak{t}) = \mathcal{VL}_{\mathfrak{m}}(4\mathfrak{t}-2).$$

Here, $\mathcal{VL}^*_{\mathfrak{m}}(\mathfrak{t})$ are generated using the following formula, with the given starting values:

$$\mathcal{VL}^*_{\mathfrak{m}}(\mathfrak{t}) = (4\mathfrak{t}-2)\,\mathcal{VL}^*_{\mathfrak{m}-1}(\mathfrak{t}) - \mathcal{VL}^*_{\mathfrak{m}-2}(\mathfrak{t}), \quad \mathfrak{m} \geq 2,$$

$$\mathcal{VL}^*_0(\mathfrak{t}) = 2, \quad \mathcal{VL}^*_1(\mathfrak{t}) = 4\mathfrak{t}-2.$$

Moreover, $\mathcal{VL}^*_{\mathfrak{m}}(\mathfrak{t})$ can be expressed analytically as follows:

$$\mathcal{VL}^*_{\mathfrak{m}}(\mathfrak{t}) = \begin{cases} 2, & \mathfrak{m}=0, \\ \sum_{k=0}^{\mathfrak{m}}(-1)^{\mathfrak{m}-k}\frac{4^k(2\mathfrak{m})(\mathfrak{m}+k-1)!}{(2k)!(\mathfrak{m}-k)!}\mathfrak{t}^k, & \mathfrak{m} \geq 1. \end{cases} \tag{7}$$

The orthogonality condition for SVLPs can be expressed as:

$$\langle \mathcal{VL}^*_{\mathfrak{m}}(\mathfrak{t}), \mathcal{VL}^*_{\imath}(\mathfrak{t})\rangle = \int_0^1 \mathcal{W}^*(\mathfrak{t})\mathcal{VL}^*_{\mathfrak{m}}(\mathfrak{t})\mathcal{VL}^*_{\imath}(\mathfrak{t})\mathrm{d}\mathfrak{t} = \begin{cases} 4\pi & \mathfrak{m} = \imath = 0, \\ 2\pi & \mathfrak{m} = \imath \neq 0, \\ 0 & \mathfrak{m} \neq \imath. \end{cases} \tag{8}$$

Where $\mathcal{W}^*(\mathfrak{t}) = \frac{1}{\sqrt{\mathfrak{t}-\mathfrak{t}^2}}$ is the weight function.

3. Operational Matrix of VO Derivative Operators

The operational matrix for the ABC fractional derivative is a key element of the spectral collocation method proposed for solving MT-VOFDEs. This matrix enables the transformation of fractional derivatives into algebraic operations using the SVLP basis. We present its construction as follows.

- **Approximate the function.** Any function $\mathrm{U}(\mathfrak{t}) \in \mathcal{L}^2[0,1]$ can be approximated using SVLPs as:

$$\mathrm{U}(\mathfrak{t}) = \sum_{k=0}^{\infty} c_k \mathcal{VL}^*_k(\mathfrak{t}).$$

If $\mathrm{U}(\mathfrak{t})$ is a polynomial of degree $\mathfrak{m}$, it can be represented in terms of SVLPs as:

$$\mathrm{U}(\mathfrak{t}) \cong \mathrm{U}_{\mathfrak{m}}(\mathfrak{t}) = \sum_{k=0}^{\mathfrak{m}} c_k \mathcal{VL}^*_k(\mathfrak{t}) = \mathcal{C}^T \vartheta(\mathfrak{t}), \tag{9}$$

where $\vartheta(\mathfrak{t}) = [\mathcal{VL}^*_0(\mathfrak{t}), \mathcal{VL}^*_1(\mathfrak{t}), \ldots, \mathcal{VL}^*_{\mathfrak{m}}(\mathfrak{t})]^T$ is the vector of SVLPs, and $\mathcal{C} = [c_0, c_1, \ldots, c_{\mathfrak{m}}]^T$ is the coefficient vector of length $\mathfrak{m}+1$. This vector can be computed using the following integration method:

$$c_k = \frac{1}{\delta_k \pi} \int_0^1 \mathrm{U}(\mathfrak{t})\mathcal{W}^*(\mathfrak{t})\mathcal{VL}^*_k(\mathfrak{t})d\mathfrak{t},$$

where

$$\delta_k = \begin{cases} 4, & k = 0, \\ 2, & k \geq 1. \end{cases} \tag{10}$$

The vector $\vartheta(\mathfrak{t})$ can be expressed in the matrix form as:

$$\vartheta(\mathfrak{t}) = \mathcal{A}\mathcal{T}(\mathfrak{t}), \tag{11}$$

where

$$\mathcal{T}(\mathfrak{t}) = \left[1, \mathfrak{t}, \mathfrak{t}^2, \ldots, \mathfrak{t}^{\mathfrak{m}}\right]^T.$$

The matrix $\mathcal{A}$ is a lower triangular square matrix of size $(\mathfrak{m}+1)(\mathfrak{m}+1)$ with elements defined as:

$$\mathcal{A} = \begin{bmatrix} a_{00} & a_{01} & \cdots & a_{0\mathfrak{m}} \\ a_{10} & a_{11} & \cdots & a_{1\mathfrak{m}} \\ \vdots & \vdots & \ddots & \vdots \\ a_{\mathfrak{m}0} & a_{\mathfrak{m}1} & \cdots & a_{\mathfrak{m}\mathfrak{m}} \end{bmatrix},$$

where the a_{ij}-elements of matrix $\mathcal{A}$ are given by:

$$a_{ij} = \begin{cases} 2, & i = j = 0, \\ (-1)^{i-j} \frac{4^j (2i)(i+j-1)!}{(2j)!(i-j)!}, & i \geq j, \\ 0, & otherwise. \end{cases}$$

For example, when $\mathfrak{m} = 4$, the square matrix $\mathcal{A}$ is denoted as:

$$\mathcal{A} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ -2 & 4 & 0 & 0 & 0 \\ 2 & -16 & 16 & 0 & 0 \\ 2 & 36 & -96 & 64 & 0 \\ 2 & -64 & 320 & -512 & 256 \end{bmatrix}.$$

- **Operational matrix for the integer-order derivative..** The integer-order derivative of vector $\vartheta(\mathfrak{t})$ can be given as:

$$\begin{aligned} \frac{d}{d\mathfrak{t}}\vartheta(\mathfrak{t}) &= \frac{d}{d\mathfrak{t}}\left(\mathcal{A}\mathcal{T}(\mathfrak{t})\right) = \mathcal{A}\frac{d}{d\mathfrak{t}}\mathcal{T}(\mathfrak{t}) \\ &= \mathcal{A}\frac{d}{d\mathfrak{t}}\left[1, \mathfrak{t}, \mathfrak{t}^2, \dots, \mathfrak{t}^{\mathfrak{m}}\right] = \mathcal{A}\left[0, 1, 2\mathfrak{t}, \dots, \mathfrak{m}\mathfrak{t}^{\mathfrak{m}-1}\right] \\ &= \mathcal{A}\begin{bmatrix} 0 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ 0 & 2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathfrak{m} \end{bmatrix}\begin{bmatrix} 1 \\ \mathfrak{t} \\ \mathfrak{t}^2 \\ \vdots \\ \mathfrak{t}^{\mathfrak{m}-1} \end{bmatrix} = \mathcal{A}I\mathcal{T}^*(\mathfrak{t}), \end{aligned}$$

where

$$I = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ 0 & 2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathfrak{m} \end{bmatrix}, \quad \text{and } \mathcal{T}^*(\mathfrak{t}) = \begin{bmatrix} 1 \\ \mathfrak{t} \\ \mathfrak{t}^2 \\ \vdots \\ \mathfrak{t}^{\mathfrak{m}-1} \end{bmatrix}.$$

From Eq. (11), we obtain:

$$\mathcal{T}(\mathfrak{t}) = \mathcal{A}^{-1}\vartheta(\mathfrak{t}). \tag{12}$$

Then

$$\mathcal{T}^*(\mathfrak{t}) = F^*\vartheta(\mathfrak{t}),$$

where

$$F^* = \begin{bmatrix} \mathcal{A}_1^{-1} \\ \mathcal{A}_2^{-1} \\ \vdots \\ \mathcal{A}_{\mathfrak{m}}^{-1} \end{bmatrix}.$$

Here, the vector $\mathcal{A}_r^{-1}$ is the row of $\mathcal{A}^{-1}$, $r = 1, 2, \dots, \mathfrak{m}$. Thus,

$$\frac{d}{d\mathfrak{t}}\vartheta(\mathfrak{t}) = D^{(1)}\vartheta(\mathfrak{t}), \tag{13}$$

where

$$D^{(1)} = \mathcal{A}IF^*.$$

For the $\mathfrak{n}$-th derivative of $\vartheta(\mathfrak{t})$, we have:

$$\frac{d^{\mathfrak{n}}}{d^{\mathfrak{n}}\mathfrak{t}}\vartheta(\mathfrak{t}) = D^{(\mathfrak{n})}\vartheta(\mathfrak{t}), \tag{14}$$

where

$$D^{(\mathfrak{n})} = (\mathcal{A}IF^*)^{(\mathfrak{n})}.$$

The matrix $D^{(\mathfrak{n})}$ is the operational matrix of integer-order derivatives by SVLPs. As a direct result, we obtain the derivative of $\mathrm{U}(\mathfrak{t})$ as:

$$\frac{d}{d\mathfrak{t}}\mathrm{U}(\mathfrak{t}) = \frac{d}{d\mathfrak{t}}(\mathcal{C}^T\vartheta(\mathfrak{t})) = \mathcal{C}^T(\frac{d}{d\mathfrak{t}}\vartheta(\mathfrak{t})) = \mathcal{C}^T D^{(1)}\vartheta(\mathfrak{t}). \tag{15}$$

- **Operational matrix for ABC variable-order derivative.**

The ABC variable-order derivative of the vector $\vartheta(\mathfrak{t})$ can be given as:

$$^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}\vartheta(\mathfrak{t}) = {}^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}\left(\mathcal{A}\mathcal{T}(\mathfrak{t})\right) = \mathcal{A}\left({}^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}\mathcal{T}(\mathfrak{t})\right) = \mathcal{A}\,{}^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}\left[1, \mathfrak{t}, \mathfrak{t}^2, \ldots, \mathfrak{t}^{\mathfrak{m}}\right].$$

From Eq. (5), we obtain

$$\begin{aligned} ^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}\vartheta(\mathfrak{t}) &= \mathcal{A}\Big[\; 0 \quad \tfrac{N(\xi_i(\mathfrak{t}))\mathfrak{t}}{1-\xi_i(\mathfrak{t})}E_{\xi_i(\mathfrak{t}),2}\left(\tfrac{-\xi_i(\mathfrak{t})}{1-\xi_i(\mathfrak{t})}\mathfrak{t}^{\xi_i(\mathfrak{t})}\right) \\ &\qquad \cdots \quad \tfrac{N(\xi_i(\mathfrak{t}))\mathfrak{t}^{\mathfrak{m}}\Gamma(\mathfrak{m}+1)}{1-\xi_i(\mathfrak{t})}E_{\xi_i(\mathfrak{t}),\mathfrak{m}+1}\left(\tfrac{-\xi_i(\mathfrak{t})}{1-\xi_i(\mathfrak{t})}\mathfrak{t}^{\xi_i(\mathfrak{t})}\right) \;\Big]^T \\ &= \mathcal{A}\mathcal{M}_i\left[1, \mathfrak{t}, \mathfrak{t}^2, \ldots, \mathfrak{t}^{\mathfrak{m}}\right] = \mathcal{A}\mathcal{M}_i\mathcal{T}(\mathfrak{t}) \end{aligned}$$

By using Eq. (12), we have

$$^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}\vartheta(\mathfrak{t}) = \Omega_i\vartheta(\mathfrak{t}), \tag{16}$$

where

$$\Omega_i = \mathcal{A}\mathcal{M}_i\mathcal{A}^{-1}.$$

The square matrices Ω_i are the operational matrices of the ABC derivative of variable order by SVLPs, and

$$\mathcal{M}_i = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & \frac{N(\xi_i(\mathfrak{t}))}{1-\xi_i(\mathfrak{t})}\sum_{k=0}^{\infty}\frac{(-\xi_i(\mathfrak{t}))^k}{(1-\xi_i(\mathfrak{t}))^k\Gamma(\xi_i(\mathfrak{t})k+2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & \frac{N(\xi_i(\mathfrak{t}))}{1-\xi_i(\mathfrak{t})}\sum_{k=0}^{\infty}\frac{(-\xi_i(\mathfrak{t}))^k\Gamma(\mathfrak{m}+1)}{(1-\xi_i(\mathfrak{t}))^k\Gamma(\xi_i(\mathfrak{t})k+\mathfrak{m}+1)} \end{bmatrix}.$$

As a direct result, we derive the operational matrix representation for $^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}\mathrm{U}(\mathfrak{t})$ as:

$$^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}\mathrm{U}(\mathfrak{t}) = {}^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}(\mathcal{C}^T\vartheta(\mathfrak{t})) = \mathcal{C}^T({}^{ABC}\mathcal{D}^{\xi_i(\mathfrak{t})}\vartheta(\mathfrak{t})) = \mathcal{C}^T\Omega_i\vartheta(\mathfrak{t}), \tag{17}$$

4. Proposed Method for Solving MT-VOFDEs

To find the solution $\mathrm{U}(\mathfrak{t})$, we use the following steps:

- Step 1. Consider Eq. (1)..
- Step 2. Approximate the unknown function $\mathrm{U}(\mathfrak{t})$ and the initial condition given in Eq. (2) using Eq. (9) and substitute in Eq. (1).
- Step 3. Compute the operational matrices from Eq. (17) and substitute them in Eq. (1).
  - The results of Steps 2 and 3 are given below

$$\sum_{i=1}^{\mathfrak{n}}\Upsilon_i(\mathfrak{t})\mathcal{C}^T\Omega_i\vartheta(\mathfrak{t}) = \mathcal{Z}(\mathfrak{t}, \mathcal{C}^T\vartheta(\mathfrak{t})), \quad 0 \le \mathfrak{t} \le 1, \tag{18}$$

$$\mathcal{C}^T\vartheta(0) = \mathrm{U}_0. \tag{19}$$

- Step 4. To solve Eq. (18) with the initial condition given in Eq. (19), we choose the collocation points as:

$$\mathfrak{t}_r = \frac{2r+1}{2\mathfrak{m}+2}, \quad r = 0, 1, \ldots, \mathfrak{m}.$$

- Replace the row corresponding to the initial condition Eq. (19) with the last row of the matrix equation and compute the coefficient vector $\mathcal{C}$ by solving the modified linear system.
- The approximate solution at any point $\mathfrak{t}$ is given by substituting vector $\mathcal{C}$ in Eq. (9).

5. Error Analysis

The error analysis quantifies how close the numerical solution is to the exact solution. We now present a theorem that provides an explicit bound on the error.

Theorem 5.1. *Let* $\mathrm{U}(\mathfrak{t}) \in L^2[0,1]$ *be a function such that* $|\mathrm{U}^{(2)}(\mathfrak{t})| \leq \mathfrak{N}$, $\forall \mathfrak{t} \in [0,1]$, *with its series expansion given by* $\mathrm{U}(\mathfrak{t}) = \sum_{k=0}^{\infty} c_k \mathcal{VL}_k^*(\mathfrak{t})$, *where* $\mathcal{VL}_{\mathfrak{m}}^*(\mathfrak{t}) = 2T_{\mathfrak{m}}(2\mathfrak{t}-1)$, *and* $T_{\mathfrak{m}}(\mathfrak{t})$ *is the Chebyshev polynomial of the first kind* [31]*. Let* $\mathrm{U}_{\mathfrak{m}}(\mathfrak{t}) = \sum_{k=0}^{\mathfrak{m}} c_k \mathcal{VL}_k^*(\mathfrak{t})$ *denote the truncated approximation obtained by the proposed method. Then the maximum error satisfies*

$$\sup_{\mathfrak{t}\in[0,1]} |\mathrm{U}(\mathfrak{t}) - \mathrm{U}_{\mathfrak{m}}(\mathfrak{t})| \leq \frac{\mathfrak{N}}{\mathfrak{m}^{\frac{3}{2}}\sqrt{96\pi}} + 2\sqrt{\mathfrak{m}+1}\left\|\tilde{C} - C\right\|_2,$$

where $C = [c_0, c_1, \ldots, c_{\mathfrak{m}}]^T$ *is the coefficient vector of the approximate solution, and* $\tilde{C} = [\tilde{c}_0, \tilde{c}_1, \ldots, \tilde{c}_{\mathfrak{m}}]^T$ *is the coefficient vector of an intermediate approximation.*

Proof. Consider an intermediate approximation $\tilde{\mathrm{U}}_{\mathfrak{m}}(\mathfrak{t}) = \sum_{k=0}^{\mathfrak{m}} \tilde{c}_k \mathcal{VL}_k^*(\mathfrak{t})$. The error can be expressed as:

$$|\mathrm{U}(\mathfrak{t}) - \mathrm{U}_{\mathfrak{m}}(\mathfrak{t})| \leq |\mathrm{U}(\mathfrak{t}) - \tilde{\mathrm{U}}_{\mathfrak{m}}(\mathfrak{t})| + |\tilde{\mathrm{U}}_{\mathfrak{m}}(\mathfrak{t}) - \mathrm{U}_{\mathfrak{m}}(\mathfrak{t})|.$$

By analogy to Theorem 1 in [30], we have:

$$|\mathrm{U}(\mathfrak{t}) - \tilde{\mathrm{U}}_{\mathfrak{m}}(\mathfrak{t})| \leq \frac{\mathfrak{N}}{\mathfrak{m}^{\frac{3}{2}}\sqrt{96\pi}}.$$

For the second term, applying the Cauchy-Schwarz inequality yields

$$|\tilde{\mathrm{U}}_{\mathfrak{m}}(\mathfrak{t}) - \mathrm{U}_{\mathfrak{m}}(\mathfrak{t})| = \left|\sum_{k=0}^{\mathfrak{m}} (\tilde{c}_k - c_k)\mathcal{VL}_k^*(\mathfrak{t})\right| \leq \sqrt{\sum_{k=0}^{\mathfrak{m}} (\tilde{c}_k - c_k)^2}\sqrt{\sum_{k=0}^{\mathfrak{m}} |\mathcal{VL}_k^*(\mathfrak{t})|^2}.$$

Since $|\mathcal{VL}_k^*(\mathfrak{t})| = |2T_k(2\mathfrak{t}-1)| \leq 2$, for all k and $\mathfrak{t} \in [0,1]$,

$$\sqrt{\sum_{k=0}^{\mathfrak{m}} |\mathcal{VL}_k^*(\mathfrak{t})|^2} \leq \sqrt{\sum_{k=0}^{\mathfrak{m}} 4} = 2\sqrt{\mathfrak{m}+1}.$$

Thus:

$$|\tilde{\mathrm{U}}_{\mathfrak{m}}(\mathfrak{t}) - \mathrm{U}_{\mathfrak{m}}(\mathfrak{t})| \leq 2\sqrt{\mathfrak{m}+1}\left\|\tilde{C} - C\right\|_2.$$

Combining these bounds, we obtain:

$$\sup_{\mathfrak{t}\in[0,1]} |\mathrm{U}(\mathfrak{t}) - \mathrm{U}_{\mathfrak{m}}(\mathfrak{t})| \leq \frac{\mathfrak{N}}{\mathfrak{m}^{\frac{3}{2}}\sqrt{96\pi}} + 2\sqrt{\mathfrak{m}+1}\left\|\tilde{C} - C\right\|_2.$$

□

6. Test examples

In this section, we present several test problems to demonstrate the efficiency, accuracy, and convergence of the proposed SVLP collocation framework for VO-ABC fractional differential equations. All computations were carried out using *Mathematica 13* on a standard desktop machine.

Example 6.1. *Let* $\Upsilon_1(\mathtt{t}) = 1, \xi_1(\mathtt{t}) = \mathtt{t}$, $\mathrm{U}(0) = 0$, *and*

$$\mathcal{Z}(\mathtt{t}, \mathrm{U}(\mathtt{t})) = \frac{N(\xi_1(\mathtt{t}))}{1-\xi_1(\mathtt{t})}\left[\mathtt{t}^2\Gamma(3)E_{\xi_1(\mathtt{t}),3}\left(\frac{-\xi_1(\mathtt{t})}{1-\xi_1(\mathtt{t})}\mathtt{t}^{\xi_1(\mathtt{t})}\right) - \mathtt{t}\Gamma(2)E_{\xi_1(\mathtt{t}),2}\left(\frac{-\xi_1(\mathtt{t})}{1-\xi_1(\mathtt{t})}\mathtt{t}^{\xi_1(\mathtt{t})}\right)\right]. \tag{20}$$

The exact solution is $\mathrm{U}(\mathtt{t}) = \mathtt{t}^2 - \mathtt{t}$.

Applying the proposed method yields the numerical results shown in Tables 1-3 and Figures 1-2. The approximate solutions remain consistent across polynomial degrees, while the absolute errors reported in Table 2 are extremely small, often near machine precision, and decrease as $\mathtt{m}$ increases. Figures 1(a,b) confirm the close agreement with the exact solution and the minimal error distribution across the domain. Error norms in Table 3 and their trends in Figure 2 further demonstrate the high accuracy, stability, and efficiency of the method, with only moderate growth in CPU time. Overall, Example 6.1 validates the robustness of the SVLPs framework for fractional-order problems.

Table 1. Approximate solution for various $\mathtt{m}$ values.

t	m = 2	m = 3	m = 4	m = 5
0.1	-0.0900	-0.09000000000000016	-0.0900	-0.09000000000000001
0.2	-0.1600	-0.16000000000000061	-0.15999999999999998	-0.16000000000000003
0.3	-0.21000000000000002	-0.2100000000000013	-0.20999999999999994	-0.21000000000000016
0.4	-0.2400	-0.24000000000000196	-0.23999999999999994	-0.24000000000000013
0.5	-0.2500	-0.2500000000000027	-0.2500000000000001	-0.2500000000000001
0.6	-0.2400	-0.2400000000000033	-0.24000000000000019	-0.24000000000000002
0.7	-0.20999999999999996	-0.21000000000000357	-0.2100000000000002	-0.20999999999999996
0.8	-0.15999999999999992	-0.16000000000000342	-0.15999999999999995	-0.16000000000000017
0.9	-0.08999999999999997	-0.0900000000000297	-0.08999999999999933	-0.09000000000000094

Table 2. Exact solutions and absolute error analysis for various $\mathtt{m}$ values.

t	Exact Solution	m = 2	m = 3	m = 4	m = 5
0.1	-0.0900	0	1.66533×10^{-16}	0	1.38778×10^{-17}
0.2	-0.1600	0	6.10623×10^{-16}	2.77556×10^{-17}	2.77556×10^{-17}
0.3	-0.21000000000000002	0	1.27676×10^{-15}	8.32667×10^{-17}	1.11022×10^{-16}
0.4	-0.2400	0	1.97065×10^{-15}	5.55112×10^{-17}	1.38778×10^{-16}
0.5	-0.2500	0	2.72005×10^{-15}	1.11022×10^{-16}	1.11022×10^{-16}
0.6	-0.2400	0	3.30291×10^{-15}	1.94289×10^{-16}	2.77556×10^{-17}
0.7	-0.20999999999999996	0	3.60822×10^{-15}	2.22045×10^{-16}	0
0.8	-0.15999999999999992	0	3.4972×10^{-15}	2.77556×10^{-17}	2.498×10^{-16}
0.9	-0.08999999999999997	0	2.9976×10^{-15}	6.38378×10^{-16}	9.71445×10^{-16}

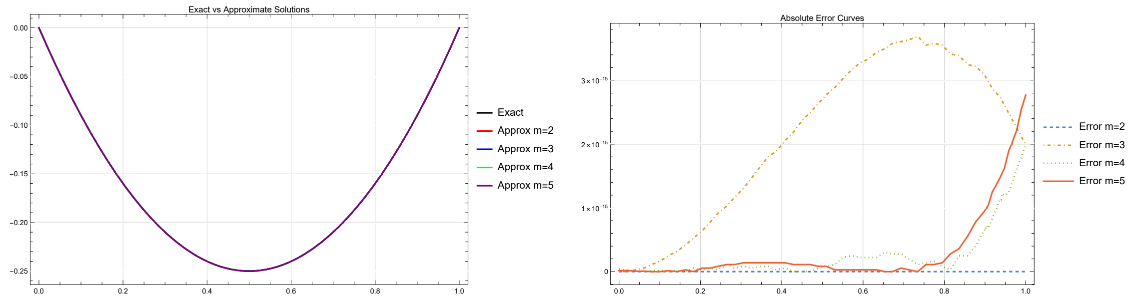
Example 6.2. *Let* $\Upsilon_1(\mathtt{t}) = 1, \xi_1(\mathtt{t}) = 1 - 0.5e^{-\mathtt{t}}$, $\mathrm{U}(0) = 2$, *and*

$$\mathcal{Z}(\mathtt{t}, \mathrm{U}(\mathtt{t})) = \frac{N(\xi_1(\mathtt{t}))}{1-\xi_1(\mathtt{t})}\mathtt{t}^4\Gamma(5)E_{\xi_1(\mathtt{t}),5}\left(\frac{-\xi_1(\mathtt{t})}{1-\xi_1(\mathtt{t})}\mathtt{t}^{\xi_1(\mathtt{t})}\right) + 2\mathrm{U}(\mathtt{t}). \tag{21}$$

The exact solution is $\mathrm{U}(\mathtt{t}) = \mathtt{t}^4 + 2$.

TABLE 3. Error norms and CPU time for Example 1

m	L^2 error	L^∞ error	CPU time (sec)
4	4.08891×10^{-16}	1.45027×10^{-15}	0.109375
6	3.8894×10^{-16}	2.0671×10^{-15}	0.171875
8	3.5691×10^{-15}	2.0461×10^{-14}	0.296875
10	2.00862×10^{-14}	1.32329×10^{-13}	0.5



(a) Exact vs Approximate Solutions for different values of m

(b) Absolute error comparison for Example 6.1 with increasing values of m

FIGURE 1. Comparison of exact and approximate solutions and corresponding absolute errors for Example 6.1.

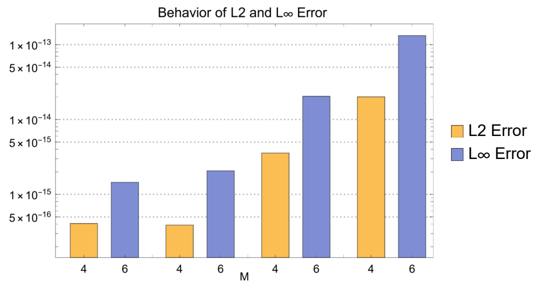


FIGURE 2. Behavior of L^2 and L^∞ Error versus m in Example 6.1.

Figure 3 displays the approximate solution and the corresponding absolute errors generated by the proposed method. Table 4 presents the absolute errors computed for different values of t. The results demonstrate that the method achieves high accuracy even with a small number of shifted Vieta–Lucas polynomials. This indicates that the method is computationally efficient while maintaining precision.

Example 6.3. *Let* $\Upsilon_1(t) = 1, \Upsilon_2(t) = 1, \xi_1(t) = \frac{1}{1+e^{5t}}, \xi_2(t) = 1$, $U(0) = 1$, *and*

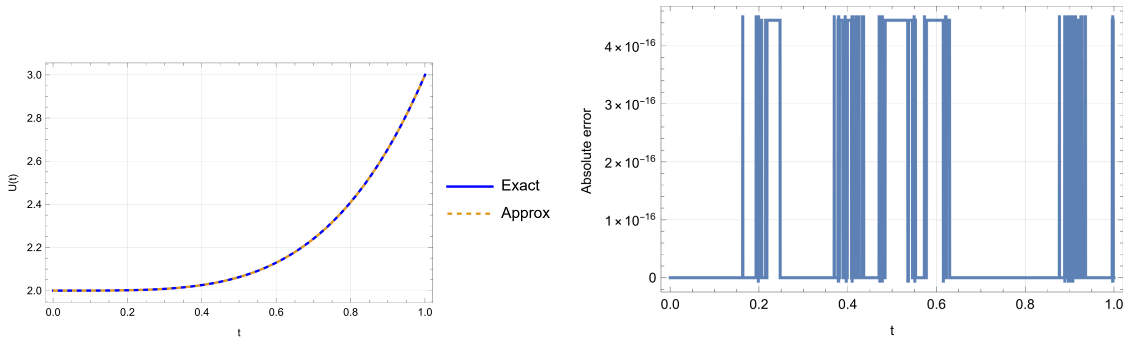
$$\mathcal{Z}(t, U(t)) = \frac{N(\xi_1(t))}{1-\xi_1(t)} \left[t^3\Gamma(4)E_{\xi_1(t),4}\left(\frac{-\xi_1(t)}{1-\xi_1(t)} t^{\xi_1(t)}\right) - t\Gamma(2)E_{\xi_1(t),2}\left(\frac{-\xi_1(t)}{1-\xi_1(t)} t^{\xi_1(t)}\right)\right] + U'(t). \tag{22}$$

The exact solution is $U(t) = t^3 - t + 1$.

Applying the proposed method, the numerical results are presented in Table 5, while Fig. 4 confirms the efficiency of the proposed method. Acceptable accuracy is achieved with low polynomial degrees, reducing the computational cost. This feature makes the method particularly suitable for large-scale or real-time applications where minimizing the number of basis functions is crucial.

Table 4. Exact and Approximate Solutions with Absolute Error

t	Exact Solution	Approximate Solution	Absolute Error
0.1	2.0000	2.0001	4.44089×10^{-16}
0.2	2.0016	2.0016	0
0.3	2.0081	2.0081	0
0.4	2.0256	2.0256	4.44089×10^{-16}
0.5	2.0625	2.0625	0
0.6	2.1296	2.1296	4.44089×10^{-16}
0.7	2.2401	2.2401	0
0.8	2.4096	2.4096	0
0.9	2.6561	2.6561	0



(a) Exact vs Approximate Solutions (b) Absolute Error when $\mathtt{m}=4$

Figure 3. Comparison of exact/approximate solutions and absolute error for Example 6.2.

Table 5. Exact and Approximate Solutions with Absolute Error

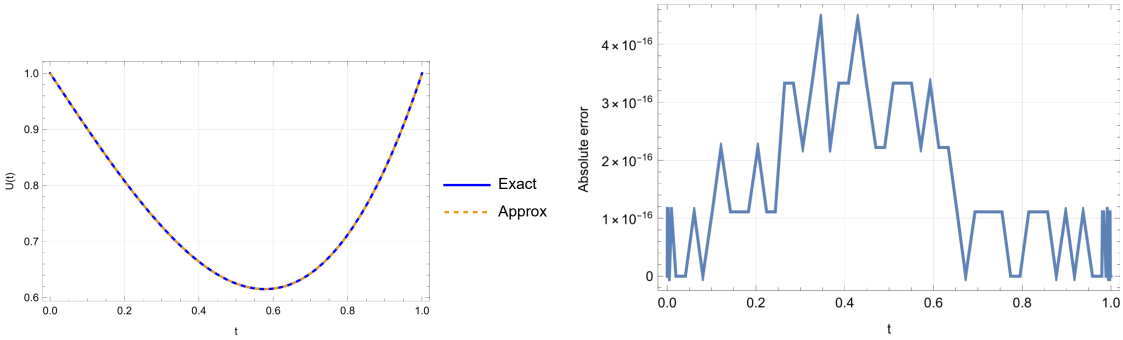
t	Exact Solution	Approximate Solution	Absolute Error
0.1	0.9010	0.9010	0
0.2	0.8080	0.8080	1.11022×10^{-16}
0.3	0.7270	0.7270	2.22045×10^{-16}
0.4	0.6640	0.6640	3.33067×10^{-16}
0.5	0.6250	0.6250	3.33067×10^{-16}
0.6	0.6160	0.6160	3.33067×10^{-16}
0.7	0.6430	0.6430	0
0.8	0.7120	0.7120	0
0.9	0.8290	0.8290	0

Example 6.4. *Consider the following example*[21]*. Let* $\Upsilon_1(\mathrm{t})=1, \Upsilon_2(\mathrm{t})=2, \xi_1(\mathrm{t})=\frac{1}{1+e^{5\mathrm{t}}}, \xi_2(\mathrm{t})=1-e^{-2\mathrm{t}}$, $\mathrm{U}(0)=0$, *and*

$$\begin{aligned} \mathcal{Z}(\mathrm{t},\mathrm{U}(\mathrm{t})) =& \frac{N(\xi_1(\mathrm{t}))}{1-\xi_1(\mathrm{t})}\mathrm{t}^2\Gamma(3)E_{\xi_1(\mathrm{t}),3}\left(\frac{-\xi_1(\mathrm{t})}{1-\xi_1(\mathrm{t})}\mathrm{t}^{\xi_1(\mathrm{t})}\right) \\ &+2\frac{N(\xi_2(\mathrm{t}))}{1-\xi_2(\mathrm{t})}\mathrm{t}^2\Gamma(3)E_{\xi_2(\mathrm{t}),3}\left(\frac{-\xi_2(\mathrm{t})}{1-\xi_2(\mathrm{t})}\mathrm{t}^{\xi_2(\mathrm{t})}\right)+4\mathrm{U}(\mathrm{t}). \end{aligned} \tag{23}$$

The exact solution is $\mathrm{U}(\mathrm{t})=\mathrm{t}^2$.

Applying the proposed method, the numerical results are shown in Table 6 and Fig. 5.

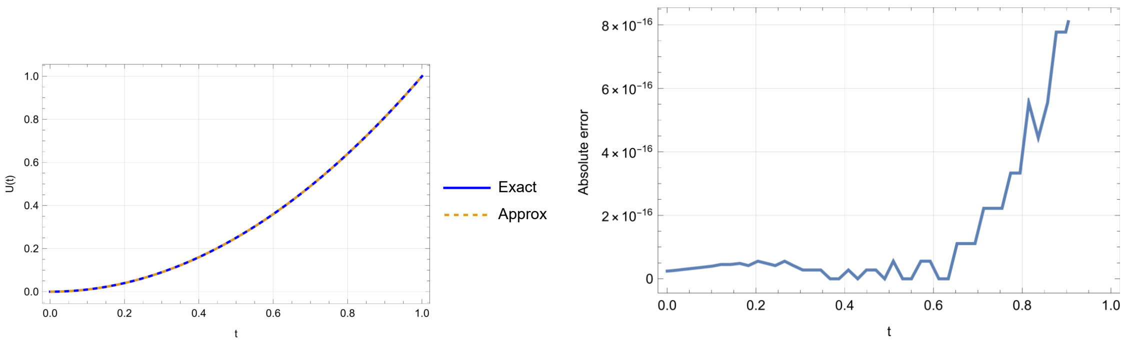


(a) Exact vs Approximate Solutions (b) Absolute Error when $\mathtt{m} = 4$

FIGURE 4. Comparison of exact/approximate solutions and absolute error for Example 6.3.

TABLE 6. Exact and Approximate Solutions with Absolute Error

t	Exact Solution	Approximate Solution	Absolute Error
0.1	0.0100	0.0100	3.98986×10^{-17}
0.2	0.0400	0.0400	4.85723×10^{-17}
0.3	0.0900	0.0900	4.16334×10^{-17}
0.4	0.1600	0.1600	0
0.5	0.2500	0.2500	2.77556×10^{-17}
0.6	0.3600	0.3600	0
0.7	0.4900	0.4900	1.11022×10^{-16}
0.8	0.6400	0.6400	3.33067×10^{-16}
0.9	0.8100	0.8100	7.77156×10^{-16}



(a) Exact vs Approximate Solutions (b) Absolute Error when $\mathtt{m} = 4$

FIGURE 5. Comparison of exact/approximate solutions and absolute error for Example 6.4.

Example 6.5. *Let* $\Upsilon_1(\mathrm{t}) = 1, \Upsilon_2(\mathrm{t}) = \mathrm{t}^{\frac{1}{2}}, \Upsilon_3(\mathrm{t}) = \mathrm{t}^{\frac{1}{3}}, \Upsilon_4(\mathrm{t}) = \mathrm{t}^{\frac{1}{4}}, \xi_1(\mathrm{t}) = \frac{\mathrm{t}}{2}, \xi_2(\mathrm{t}) = \frac{\mathrm{t}}{3}, \xi_3(\mathrm{t}) = \frac{\mathrm{t}}{4}, \xi_4(\mathrm{t}) = \frac{\mathrm{t}}{5}$, $\mathrm{U}(0) = 2$, *and*

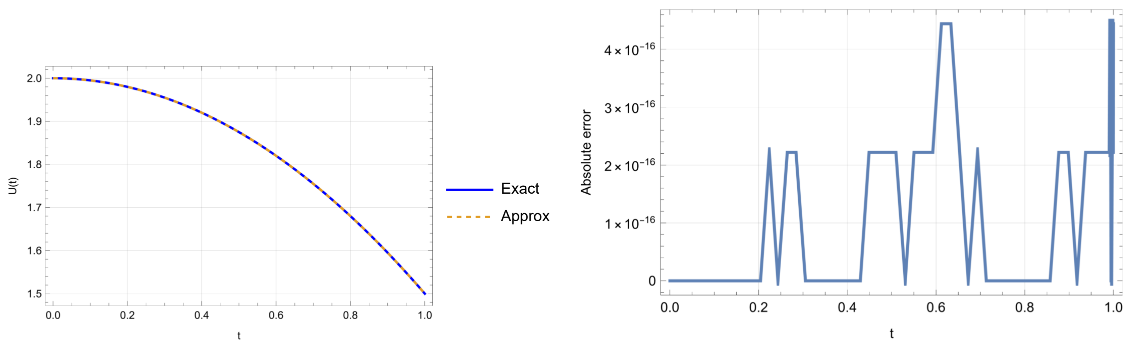
$$\begin{aligned}\mathcal{Z}(\mathrm{t}, \mathrm{U}(\mathrm{t})) = &-\frac{N(\xi_1(\mathrm{t}))}{1-\xi_1(\mathrm{t})}\mathrm{t}^2\Gamma(3)E_{\xi_1(\mathrm{t}),3}\left(\frac{-\xi_1(\mathrm{t})}{1-\xi_1(\mathrm{t})}\mathrm{t}^{\xi_1(\mathrm{t})}\right)\\ &-\mathrm{t}^{\frac{1}{2}}\frac{N(\xi_2(\mathrm{t}))}{1-\xi_2(\mathrm{t})}\mathrm{t}^2\Gamma(3)E_{\xi_2(\mathrm{t}),3}\left(\frac{-\xi_2(\mathrm{t})}{1-\xi_2(\mathrm{t})}\mathrm{t}^{\xi_2(\mathrm{t})}\right)\\ &-\mathrm{t}^{\frac{1}{3}}\frac{N(\xi_3(\mathrm{t}))}{1-\xi_3(\mathrm{t})}\mathrm{t}^2\Gamma(3)E_{\xi_3(\mathrm{t}),3}\left(\frac{-\xi_3(\mathrm{t})}{1-\xi_3(\mathrm{t})}\mathrm{t}^{\xi_3(\mathrm{t})}\right)\\ &-\mathrm{t}^{\frac{1}{4}}\frac{N(\xi_4(\mathrm{t}))}{1-\xi_4(\mathrm{t})}\mathrm{t}^2\Gamma(3)E_{\xi_4(\mathrm{t}),3}\left(\frac{-\xi_4(\mathrm{t})}{1-\xi_4(\mathrm{t})}\mathrm{t}^{\xi_4(\mathrm{t})}\right)\\ &+\mathrm{t}^{\frac{1}{5}}\mathrm{U}(\mathrm{t}).\end{aligned} \tag{24}$$

The exact solution is $\mathrm{U}(\mathrm{t}) = 2 - \frac{\mathrm{t}^2}{2}$.

Applying the proposed method, the numerical results are shown in Table 7 and Fig. 6.

TABLE 7. Exact and Approximate Solutions with Absolute Error

t	Exact Solution	Approximate Solution	Absolute Error
0.1	1.9950	1.9950	2.22045×10^{-16}
0.2	1.9800	1.9800	0
0.3	1.9550	1.9550	2.22045×10^{-16}
0.4	1.9200	1.9200	0
0.5	1.8750	1.8750	2.22045×10^{-16}
0.6	1.8200	1.8200	2.22045×10^{-16}
0.7	1.7550	1.7550	0
0.8	1.6800	1.6800	0
0.9	1.5950	1.5950	2.22045×10^{-16}



(a) Exact vs Approximate Solutions (b) Absolute Error when $\mathtt{m} = 2$

FIGURE 6. Comparison of exact/approximate solutions and absolute error for Example 6.5.

Example 6.6. *We consider the coupled system* [32].

$$\begin{aligned} {}^{ABC}\mathcal{D}_0^{\xi(\mathrm{t})}\mathrm{U}_1(\mathrm{t}) - \mathrm{U}_1(\mathrm{t}) - \mathrm{U}_2(\mathrm{t}) &= \mathcal{Z}_1(\mathrm{t}), \\ {}^{ABC}\mathcal{D}_0^{\xi(\mathrm{t})}\mathrm{U}_2(\mathrm{t}) - \mathrm{U}_1(\mathrm{t}) - \mathrm{U}_2(\mathrm{t}) &= \mathcal{Z}_2(\mathrm{t}), \end{aligned} \tag{25}$$

with $\xi(\mathrm{t}) = 1 - \frac{1}{2}e^{-\mathrm{t}}$, $\mathrm{U}_1(0) = 0$, $\mathrm{U}_2(0) = 0$, *and*

$$\mathcal{Z}_1(\mathrm{t}) = \frac{N(\xi(\mathrm{t}))}{1-\xi(\mathrm{t})}\mathrm{t}^5\Gamma(6)E_{\xi(\mathrm{t}),6}\left(\frac{-\xi(\mathrm{t})}{1-\xi(\mathrm{t})}\mathrm{t}^{\xi(\mathrm{t})}\right) - \mathrm{t}^5 - \mathrm{t}^4,$$

$$\mathcal{Z}_2(\mathrm{t}) = \frac{N(\xi(\mathrm{t}))}{1-\xi(\mathrm{t})}\mathrm{t}^4\Gamma(5)E_{\xi(\mathrm{t}),5}\left(\frac{-\xi(\mathrm{t})}{1-\xi(\mathrm{t})}\mathrm{t}^{\xi(\mathrm{t})}\right) - \mathrm{t}^5 - \mathrm{t}^4.$$

The exact solutions are

$$\mathrm{U}_1(\mathrm{t}) = \mathrm{t}^5, \quad \mathrm{U}_2(\mathrm{t}) = \mathrm{t}^4.$$

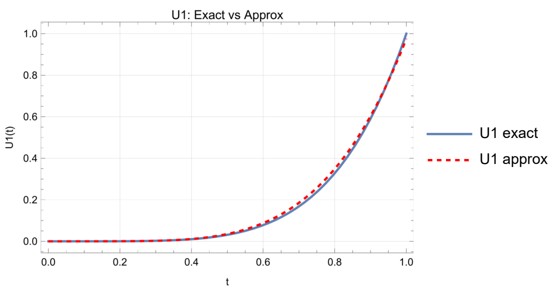
Tables 8 and 9 show that the approximate solutions of $\mathrm{U}_1(\mathrm{t})$ and $\mathrm{U}_2(\mathrm{t})$ closely match the exact analytical forms, with absolute errors remaining small across the domain. Figures 7(a) and 8(a) confirm this agreement, as the numerical and exact solutions nearly overlap. The error plots in Figures 7(b) and 8(b) further demonstrate that deviations are minimal, even as t increases. These results indicate that the proposed SVLPs framework is accurate, stable, and efficient for coupled fractional systems with variable-order ABC derivatives.

TABLE 8. Exact and Approximate Solutions with Absolute Error of $U_1(t)$ for Example 6.6

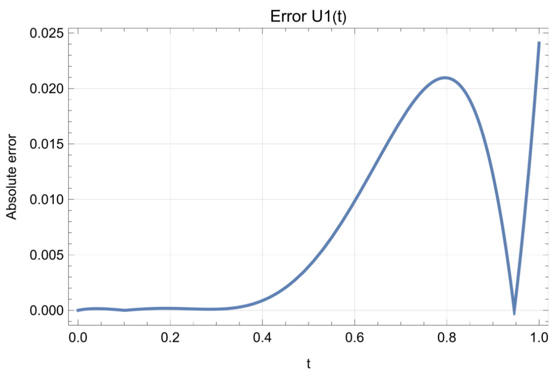
t	Exact Solution	Approximate Solution	Absolute Error
0.1	0.00001	0.0000051321	4.8679×10^{-6}
0.2	0.00032	0.000496031	1.76031×10^{-4}
0.3	0.00243	0.00253478	1.04775×10^{-4}
0.4	0.01024	0.0111194	8.79389×10^{-4}
0.5	0.03125	0.0351838	3.93384×10^{-3}
0.6	0.07776	0.0875981	9.83806×10^{-3}
0.7	0.16807	0.185168	1.70979×10^{-2}
0.8	0.32768	0.348635	2.09552×10^{-2}
0.9	0.59049	0.602678	1.21877×10^{-2}

TABLE 9. Exact and Approximate Solutions with Absolute Error of $U_2(t)$ for Example 6.6

t	Exact Solution	Approximate Solution	Absolute Error
0.1	0.0001	0.000134156	3.41564×10^{-5}
0.2	0.0016	0.00143045	1.69546×10^{-4}
0.3	0.0081	0.00812156	2.15608×10^{-5}
0.4	0.0256	0.0268212	1.22116×10^{-3}
0.5	0.0625	0.066524	4.02395×10^{-3}
0.6	0.1296	0.138606	9.00565×10^{-3}
0.7	0.2401	0.256823	1.6723×10^{-2}
0.8	0.4096	0.437314	2.77137×10^{-2}
0.9	0.6561	0.698597	4.24965×10^{-2}



(a) Exact vs Approximate for $U_1(t)$

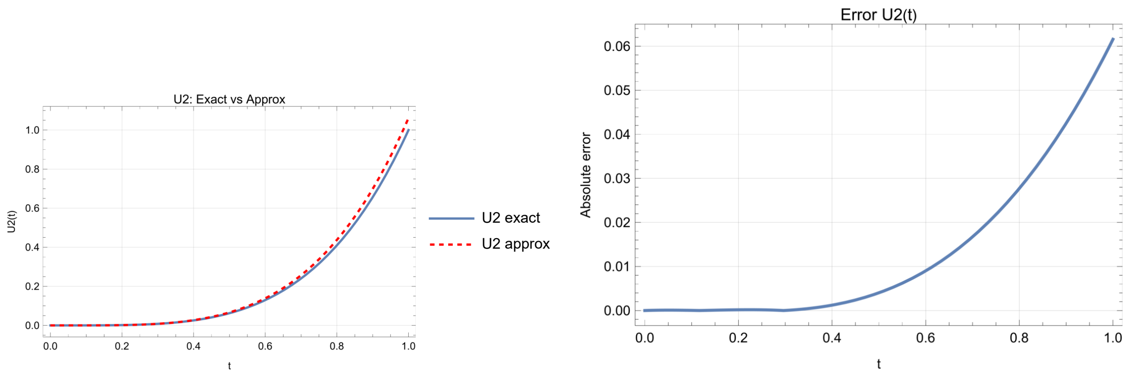


(b) Absolute Error when $m = 4$

FIGURE 7. Comparison of exact/approximate solution and absolute error of $U_1(t)$ for Example 6.6.

7. CONCLUSION

In this work, we proposed a novel spectral collocation method based on SVLPs for the numerical approximation of MT-VOFDEs involving the ABC fractional derivative. By deriving an operational matrix for the ABC operator in the SVLPs framework, the fractional system was efficiently transformed into an algebraic system, enabling rapid and accurate computation.



(a) Exact vs Approximate for $\mathrm{U}_2(\mathrm{t})$ (b) Absolute Error when $\mathtt{m} = 4$

FIGURE 8. Comparison of exact/approximate solution and absolute error of $\mathrm{U}_2(\mathrm{t})$ for Example 6.6.

The proposed method offers several key advantages: spectral accuracy, low computational cost, and numerical stability. Through test examples with known exact solutions, we demonstrated the exponential convergence and high precision of the approach, even for low polynomial degrees. Compared with existing spectral methods based on polynomials, the SVLP-based method exhibited improved error performance and computational efficiency.

Importantly, this study contributes to the ongoing development of reliable and efficient numerical techniques for solving variable-order fractional models with non-singular memory kernels. The use of SVLPs in the context of ABC-type derivatives opens new possibilities for handling more complex and realistic fractional models.

In future work, the method may be extended to systems involving spatial variables (partial differential equations), non-linear fractional models, and real-world applications in viscoelasticity, anomalous diffusion, and bioengineering. Further investigation into adaptive schemes and GPU-based parallel implementations would also enhance the method's applicability to large-scale simulations.

References

[1] Villa-Morales, J., Rincón, L., and Becerra-Guzmán, G., (2025), Optimal control of linear fractional differential equations in the Caputo sense, J. Comput. Appl. Math., 116707.

[2] Monroy, D. A., and Raposo, E. P., (2024), Solution of the space-fractional diffusion equation on bounded domains of superdiffusive phenomena, Physical Review E, 110(5), 054119.

[3] Boulaaras, S., Jan, R., and Pham, V. T., (2023), Recent advancement of fractional calculus and its applications in physical systems, Eur. Phys. J. Spec. Top., 232(14), 2347-2350.

[4] Rossikhin, Y. A., and Shitikova, M. V., (1997), Application of fractional derivatives to the analysis of damped vibrations of viscoelastic single mass systems, Acta Mechanica, 120(1), 109-125.

[5] Podlubny, I., (1998), Fractional differential equations: an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications (Vol. 198), Elsevier.

[6] Salah, E. Y., Sontakke, B., Abdo, M. S., Shatanawi, W., Abodayeh, K., and Albalwi, M. D., (2024), Conformable Fractional-Order Modeling and Analysis of HIV/AIDS Transmission Dynamics, Int. J. Differ. Equ., 2024(1), 1958622.

[7] Caputo, M., and Fabrizio, M., (2015), A new definition of fractional derivative without singular kernel, Prog. Fract. Differ. Appl., 1(2), 73-85.

[8] Atangana, A., and Baleanu, D., (2016), New fractional derivatives with nonlocal and non-singular kernel: theory and application to heat transfer model, arXiv preprint arXiv:1602.03408.

[9] Ahmed, W. F., Salamooni, A. Y., and Pawar, D. D., (2022), Solution of fractional kinetic equation for Hadamard type fractional integral via Mellin transform, Gulf Journal of Mathematics, 12(1), 15-27.

[10] Ahmed, W. F., Pawar, D. D., and Patil, W. D., (2024), Solution of Fractional Kinetic Equations Involving Laguerre Polynomials via Sumudu Transform, Journal of Function Spaces, 2024(1), 9935562.

[11] Ahmed, W. F., and Pawar, D. D., (2024), Solution of Some Non-homogeneous Fractional Integral Equations by Aboodh Transform. Journal of Nonlinear Modeling and Analysis, 6(4), 1186-1199.

[12] Baleanu, D., Jajarmi, A., and Hajipour, M., (2017), A new formulation of the fractional optimal control problems involving Mittag–Leffler nonsingular kernel, Journal of Optimization Theory and Applications, 175(3), 718-737.

[13] Akgül, A., (2018), A novel method for a fractional derivative with non-local and non-singular kernel, Chaos, Solitons & Fractals, 114, 478-482.

[14] Hasan, S., El-Ajou, A., Hadid, S., Al-Smadi, M., and Momani, S., (2020), Atangana-Baleanu fractional framework of reproducing kernel technique in solving fractional population dynamics system, Chaos, Solitons & Fractals, 133, 109624.

[15] Farman, M., Akgül, A., Nisar, K. S., Ahmad, D., Ahmad, A., Kamangar, S., and Saleel, C. A., (2022), Epidemiological analysis of fractional order COVID-19 model with Mittag-Leffler kernel, AIMS Math, 7(1), 756-783.

[16] Izadi, M., and Atangana, A., (2024), Computational analysis of a class of singular nonlinear fractional multi-order heat conduction model of the human head, Scientific Reports, 14(1), 3466.

[17] Sene, N., and Abdelmalek, K., (2019), Analysis of the fractional diffusion equations described by Atangana-Baleanu-Caputo fractional derivative, Chaos, Solitons & Fractals, 127, 158-164.

[18] Samko, S. G., and Ross, B., (1993), Integration and differentiation to a variable fractional order, Integral Transforms and Special Functions, 1(4), 277-300.

[19] Li, X., and Wu, B., (2017), A new reproducing kernel method for variable order fractional boundary value problems for functional differential equations, J. Comput. Appl. Math., 311, 387-393.

[20] El-Sayed, A. A., Baleanu, D., and Agarwal, P., (2020), A novel Jacobi operational matrix for numerical solution of multi-term variable-order fractional differential equations, Journal of Taibah University for Science, 14(1), 963-974.

[21] Tural-Polat, S. N., and Dincel, A. T., (2022), Numerical solution method for multi-term variable order fractional differential equations by shifted Chebyshev polynomials of the third kind, Alexandria Engineering Journal, 61(7), 5145-5153.

[22] Amin, A. Z., Lopes, A. M., and Hashim, I., (2023), A space-time spectral collocation method for solving the variable-order fractional Fokker-Planck equation, J. Appl. Anal. Comput., 13, 969-985.

[23] Sharma, R., and Rajeev, (2025), A numerical study for three-dimensional fractional advection–diffusion equation of variable order with modified Atangana–Baleanu–Caputo derivative, Zeitschrift für angewandte Mathematik und Physik, 76(3), 103.

[24] Awad, Y. A. R., (2025), Some results of the nonlinear hybrid fractional differential equations in Banach algebra, TWMS Journal of Applied and Engineering Mathematics.

[25] Nikam, S., and Manjarekar, S., (2025), Variable order RL fractional calculus and its Applications, Journal of Fractional Calculus and Nonlinear Systems, 6(1), 29-44.

[26] Rahou, W., Salim, A., Lazreg, J. E., and Benchohra, M., (2025), Existence Results for Implicit Fractional Differential Equations with Riesz-Caputo Derivative, Journal of Fractional Calculus and Nonlinear Systems, 6(1), 45-66.

[27] Karaoglan, M., and Erdal, B. A. S., (2025), First Step to Spectral Theory with Generalized M Derivative and Applications, Journal of Fractional Calculus and Nonlinear Systems, 6(1), 67-80.

[28] Baleanu, D., and Fernandez, A., (2018), On some new properties of fractional derivatives with Mittag-Leffler kernel, Communications in Nonlinear Science and Numerical Simulation, 59, 444-462.

[29] Agarwal, P., and El-Sayed, A. A., (2020), Vieta–Lucas polynomials for solving a fractional-order mathematical physics model, Advances in Difference Equations, 2020(1), 626.

[30] Noor, Z. A., Talib, I., Abdeljawad, T., and Alqudah, M. A., (2022), Numerical study of Caputo fractional-order differential equations by developing new operational matrices of Vieta–Lucas polynomials, Fractal and Fractional, 6(2), 79.

[31] Tasci, D., and Yalcin, F., (2013), Vieta-Pell and Vieta-Pell-Lucas polynomials, Advances in Difference Equations, 2013(1), 224.

[32] Shah, K., Abdalla, B., Abdeljawad, T., and Suwan, I., (2023), An efficient matrix method for coupled systems of variable fractional order differential equations, Thermal Science, 27(Spec. issue 1), 195-210.

Ghadah Sultan Esmail Noman received her B.Sc. degree in Mathematics from the Faculty of Applied Sciences, Taiz University, Yemen, and her M.Sc. degree in Mathematics from Dr. Babasaheb Ambedkar Marathwada University, India. She is currently pursuing her Ph.D. in Mathematics at Swami Ramanand Teerth Marathwada University, India. Her primary research interests include fractional calculus, spectral methods, and the numerical solution of fractional differential equations. Her work focuses on advancing efficient computational techniques for fractional-order systems and their applications.

Prof. D. D. Pawar is the Director and Professor at the School of Mathematical Sciences, Swami Ramanand Teerth Marathwada University, Nanded, India. His research spans general relativity, cosmological models, alternative and modified theories of gravitation, and fractional calculus. He has published extensively in reputed international journals and contributed to advancing both mathematical physics and applied fractional differential equations. Prof. Pawar has supervised numerous doctoral students and is actively involved in promoting interdisciplinary research in mathematics and physics. He regularly participates in international conferences, workshops, and academic collaborations.