Journal of Mathematical Modeling
Vol. -, No. -, 2026, pp. -. Research Article

JMM

An efficient numerical method for solving time-fractional differential equations using least squares support vector regression

Reza Masoumzadeh[†], **Nader Biranvand**[‡*]

[†]*Department of Applied Mathematics, Faculty of Mathematics and Computer Sciences, Amirkabir University of Technology (Tehran Polytechnic), No. 424, Hafez Ave., Tehran 15914, Iran*
[‡]*Department of Mathematics, Faculty of Basic Sciences, Imam Ali University, Tehran, Iran*

Email(s): reza.masoumzade@aut.ac.ir, nabiranvand@gmail.com

Abstract. In this paper, a numerical method is developed for solving time-fractional differential equations. By applying the weighted and shifted Grünwald-Letnikov approximation, the time-fractional diffusion equation is transformed into a semi-discrete scheme. Subsequently, least squares support vector regression (LS-SVR) and spectral methods are applied to solve the resulting semi-discrete scheme. The collocation LS-SVR approach is proposed for training the network. The resulting optimization problem is subsequently simplified to a linear system.

Keywords: Time-fractional differential equations, least squares support vector regression, collocation LS-SVR, Hermite functions, weighted and shifted Grünwald-Letnikov approximation.
AMS Subject Classification 2010: 35R11, 65K05, 65N35

1 Introduction

Differential equations commonly arise in mathematical models used to describe various phenomena across a broad spectrum of scientific and engineering applications. Partial differential equations (PDEs) are used to model a diverse range of physical, biological, and engineering phenomena where quantities vary continuously over space and time. While classical PDEs are extremely useful for modeling systems that exhibit local, smooth changes, they often fail to capture more complex behaviors like memory effects, fractal geometries, or non-local interactions. Accordingly, fractional partial differential equations (FPDEs) can be used. The FPDEs generalize classical PDEs by incorporating fractional-order derivatives in time or space. These fractional derivatives enable more precise and adaptable modeling of complex

*Corresponding author
Received: 11 December 2025/ Revised: 24 May 2026/ Accepted: 25 May 2026
DOI: 10.22124/jmm.2026.32521.2951

systems characterized by non-local behavior, either spatially or temporally, as well as memory effects, which cannot be captured by integer-order PDEs. Memory or non-locality is one of the most important concepts in FPDEs. In classical PDEs, the solution at any point in time depends only on the current state. However, FPDEs account for memory effects or non-local interactions, where the future state depends on the entire history of the system or spatial influences over a larger domain. So their flexibility and broad applicability make them valuable in fields like physics, engineering, biology, finance, and environmental science. The foundational concepts and practical uses of fractional differential equations (FDEs) have been extensively explored in the literature, including works such as [28,43,64]. The numerical methods for fractional calculus are developed in [7, 29]. The analysis of existence, uniqueness, and structural stability of solutions to nonlinear FDEs is conducted in [17]. A physics-informed neural network approach is introduced in [51] to address time-varying impulsive FDEs without relying on labeled data. In [3], Abbaszadeh et al. proposed an innovative numerical method for solving two-dimensional FPDEs on irregular geometries by integrating machine learning techniques with an improved MLS approximation. A numerical technique integrating the finite element method (FEM) and an interpolating element-free Galerkin method for solving time-fractional diffusion-wave phenomena in two spatial dimensions over irregular domains was introduced in [16]. Authors of [34] developed a numerical technique to solve the time-fractional Sobolev equation, using the enriched Galerkin method, which combines aspects of both continuous and discontinuous Galerkin methods. Biranvand et al. [9] proposed a numerical solution technique based on energy estimates for a class of impulsive fractional advection–dispersion equations arising in anomalous diffusion. The linear and nonlinear fuzzy impulsive FDEs have been solved using the reproducing kernel Hilbert space method [36]. A local meshless numerical technique for solving Galilei-invariant fractional advection–diffusion equations was proposed in [10]. The compact local integrated radial basis functions (RBFs) technique is introduced in [1] to simulate a diffusion-wave system with the fourth-order time-fractional component. A novel spectral method utilizing fractional hybrid Jacobi functions is presented for solving a wide range of FDEs [8]. Pell operational matrix approach is applied in [50] to estimate the numerical solution of the stochastic fractional differential equation. In [2], the authors proposed a robust numerical method for solving time-fractional advection–diffusion equations with distributed-order derivatives. A fast high-order numerical algorithm is introduced in [58] for solving the time-fractional Swift-Hohenberg model. A discrete least squares meshless collocation approach for handling nonlinear multi-term time-FDEs was proposed in [26]. For solving the time-fractional Huxley equation, authors of [48] introduced a new fractional physics-informed neural networks (FPINNs). A numerical method for the valuation of American and European options, grounded in the time-fractional Black–Scholes equation, was developed in [37]. Authors of [30] developed numerical methods to solve two-dimensional time-fractional Allen-Cahn equation. Authors of [19] established a robust and computationally efficient numerical scheme for solving nonlinear distributed-order space–time fractional PDEs. A meshless numerical solution for the time-fractional diffusion-wave equation is proposed and analyzed using a generalized finite difference method (FDM) in [44]. A temporal second-order fast finite difference scheme, incorporating different boundary conditions, is developed in [42] to numerically solve time-fractional generalized Burgers' equations. In [65], a fast second-order predictor–corrector scheme was proposed for the numerical solution of the nonlinear time-fractional Benjamin–Bona–Mahony–Burgers equation. In [54], a PINN framework incorporating time-fractional derivatives was developed to extract key physical parameters from noisy and sparse datasets. Authors of [45] proposed a novel numerical framework that extends high-order approximation formulas for handling distributed-order fractional derivatives. A FDM for solving the time-fractional Liouville–Caputo and

space-Riesz fractional diffusion equation is presented in [24]. In [21], a novel homotopy perturbation method is introduced for solving a fractional-order nonlinear cable equation. This approach linearizes the nonlinear equation at each iteration, after which the resulting problems are solved via the separation of variables method. In [23], a modified variational iteration method based on He's formulation is developed for solving FDEs. In [25], a numerical scheme for solving the nonlinear fractional differential equation is developed using Chebyshev cardinal functions. The authors of [5] introduced a novel and unconditionally stable numerical method to solve the fractional time advection-dispersion equation. A new numerical approach, referred to as flatlet oblique multiwavelets, was introduced by [22] to solve fractional-order stochastic integro-differential equations. Authors of [60] introduced a Galerkin spectral element method for the solution of a fractional diffusion equation. In [49], a numerical approach leveraging cubic B-splines is effectively developed for solving the time-fractional Black–Scholes European option pricing model. A new fourth-order finite difference weighted essentially non-oscillatory (WENO) scheme for FDEs is developed in [4]. In [6], a numerical algorithm is proposed for solving an inverse problem of a time-fractional inverse heat conduction equation featuring an unknown nonlinear boundary function.

In recent years, the use of machine learning algorithms has significantly expanded, becoming a preferred approach for addressing a wide range of complex problems across various disciplines, including healthcare, finance, autonomous vehicles, image and video analysis and cybersecurity. Support vector machine (SVM) is a supervised machine learning algorithm, which operates by identifying the optimal hyperplane that maximally separates data points of different classes was developed by Corinna Cortes and Vladimir Vapnik in the early 1990s, building on earlier theoretical work, to provide an effective machine learning algorithm for binary classification and regression tasks [14]. Least squares SVM (LS-SVM) classifiers were developed by Johan A. K. Suykens and Joos Vandewalle in the late 1990s. The LS-SVM is a modification of the standard SVM that simplifies the optimization problem by converting it into a set of linear equations, making it computationally more efficient for certain applications [53]. Least squares support vector regression (LS-SVR) is a machine learning technique that extends support vector machines to regression tasks by minimizing a least squares error function, making it efficient for predicting continuous variables while maintaining a balance between model complexity and prediction accuracy. The LS-SVR has a variety of applications across different fields due to its efficiency in handling regression tasks, for instance in signal processing, time series prediction, biomedical engineering, numerical optimization and solving differential equations. The LS-SVR is applied to solve differential equations on unbounded domains in [38]. New methodologies have been introduced for solving Fredholm and Volterra integral equations numerically in [39, 41]. Authors of [40] proposed a numerical approach for simulating Volterra-Fredholm integral equations using LS-SVR. Authors of [15] proposed a novel numerical algorithm for systems of fractional Fredholm-Volterra integro-differential equations, utilizing a hybrid approach that combines LS-SVR, the Ritz method, and piecewise shifted Vieta-Lucas functions. In [11], the authors proposed a novel approach to optimal control problems by employing LS-SVR. The method utilizes Legendre basis functions to perform feature mapping into a higher-dimensional space. A machine learning algorithm based on LS-SVR is developed in [33] with the aim of numerically simulating the time-fractional KdV-Burgers equations.

Spectral methods represent a family of highly accurate numerical approaches commonly applied to the solution of differential equations, especially PDEs. These methods represent the solution to a problem as a series of smooth basis functions, typically using Fourier series, Chebyshev polynomials, or other orthogonal polynomials. The key feature of spectral methods is that they achieve very high accu-

racy, especially for problems with smooth solutions, often requiring fewer grid points or basis functions compared to other methods like finite difference or finite element methods. A comprehensive collection of algorithms, analyses, and applications of spectral methods is represented in [12, 20, 47, 56]. Spectral methods comprise the collocation, Galerkin, Petrov-Galerkin and Tau approaches. The spectral collocation method is a numerical technique that approximates solutions to differential equations by enforcing the equations to hold at a set of discrete collocation points, typically using global basis functions like polynomials or Fourier series, providing high accuracy for both problems. Spectral method-based FPINNs were proposed by the authors of [61] as a solution for FPDEs. In [57], a new collocation method is introduced to efficiently and accurately evaluate the two-dimensional elliptic PDEs. An efficient approach for solving Hadamard-type FDEs is formulated in [63] using a spectral collocation method, with mapped Jacobi logarithmic orthogonal functions as the basis functions. A new spectral method is proposed in [59] to study the stability behavior and provide numerical solutions for systems that involve stochasticity, fractional derivatives, and time delays. The spectral collocation method is applied in [27] to solve the space fractional KdV equation as well as the space fractional KdVB equation. In [31], Marzban et al. introduced an innovative spectral collocation approach for the investigation of nonlinear control systems with delay governed by fractional mixed Volterra–Fredholm integral equations. In [46], an efficient numerical scheme is presented for the 2D time-fractional telegraph equation. The proposed method couples the Laplace transform with a Chebyshev-based spectral collocation method to achieve high accuracy. An efficient method based on the shifted Chebyshev collocation spectral method for approximating the solution of the fractional model of human T-cell lymphotropic virus I (HTLV-I) was proposed by the authors of [35].

The organization of the paper is as follows. Section 2 outlines the essential background material, covering fundamental concepts of Hermite functions, spectral methods, fractional calculus, and LS-SVR. In Section 3, we propose a time-discrete scheme for discretizing the fractional derivative in time-FDEs and derive a semi-discrete formulation. Section 4 presents a numerical approach for solving the resulting semi-discrete formulation. In Section 5, we present numerical results for several problems. The paper concludes with a discussion presented in Section 6.

2 Preliminaries

In this section, the basic concepts of Hermite functions, spectral methods, fractional calculus and LS-SVR are presented.

2.1 Hermite functions

Hermite polynomials are a fundamental set of orthogonal polynomials with significant applications in probability theory, physics, and numerical analysis. Hermite polynomials $\mathcal{H}_n(\mathrm{x})$, defined over the entire real line $\mathbb{R}$, satisfy the following recurrence relation:

$$\mathcal{H}_{n+1}(\mathrm{x}) = 2\mathrm{x}\mathcal{H}_n(\mathrm{x}) - 2n\mathcal{H}_{n-1}(\mathrm{x}), \quad n \geq 1, \tag{1}$$

with the initial members of the Hermite polynomial sequence

$$\mathcal{H}_0(\mathrm{x}) = 1,$$

$$\mathcal{H}_1(\mathrm{x}) = 2\mathrm{x}.$$

Hermite functions are closely related to Hermite polynomials and are widely used in quantum mechanics, signal processing, and numerical analysis. They are defined as Hermite polynomials multiplied by a Gaussian function, providing orthogonal basis functions that are highly suitable for problems involving the Gaussian weight or for use in spectral methods. Therefore, the Hermite functions are defined by

$$\hat{\mathcal{H}}_n(\mathrm{x}) = \frac{1}{\pi^{\frac{1}{4}}\sqrt{2^n n!}} e^{-\frac{\mathrm{x}^2}{2}} \mathcal{H}_n(\mathrm{x}), \quad \mathrm{x} \in \mathbb{R}, \, n \geq 0, \tag{2}$$

such that

$$\int_{-\infty}^{+\infty} \hat{\mathcal{H}}_m(\mathrm{x}) \hat{\mathcal{H}}_n(\mathrm{x}) d\mathrm{x} = \delta_{mn},$$

where δ_{mn} represents the Kronecker delta, defined as follows:

$$\delta_{mn} = \begin{cases} 1, & m = n, \\ 0, & m \neq n. \end{cases}$$

Hermite functions can be recursively generated using the following relation:

$$\hat{\mathcal{H}}_{n+1}(\mathrm{x}) = \mathrm{x}\sqrt{\frac{2}{n+1}} \hat{\mathcal{H}}_n(\mathrm{x}) - \sqrt{\frac{n}{n+1}} \hat{\mathcal{H}}_{n-1}(\mathrm{x}), \quad n \geq 1, \tag{3}$$

with the first two Hermite functions

$$\hat{\mathcal{H}}_0(\mathrm{x}) = \pi^{-\frac{1}{4}} e^{-\frac{\mathrm{x}^2}{2}},$$
$$\hat{\mathcal{H}}_1(\mathrm{x}) = \sqrt{2}\mathrm{x}\hat{\mathcal{H}}_0(\mathrm{x}).$$

2.2 Spectral methods

Numerical techniques for solving PDEs are generally classified into local and global approaches. The FDM and FEM rely on local principles, whereas spectral methods are inherently global. In practical applications, FEMs are especially effective for problems involving complex geometries, while spectral methods provide higher accuracy, albeit with less flexibility in handling diverse domain shapes.

Spectral methods, within the framework of numerical methods for differential equations, belong to the class of weighted residual methods (WRMs), which have traditionally served as the foundation for several numerical techniques, including the FEM, finite volume method (FVM), boundary element method (BEM), and spectral methods [18]. The WRMs comprise a specific group of approximation techniques designed to minimize residuals (errors) in a particular way, leading to methods such as collocation, Galerkin, Petrov-Galerkin, and tau formulations.

First, let us consider a differential equation in its general operator form given by

$$\mathcal{L}\mathrm{u}(\mathrm{x}) = \mathrm{f}(\mathrm{x}), \quad \mathrm{x} \in \Omega, \tag{4}$$

where $\mathcal{L}$ denotes a differential operator, u represents the unknown function to be determined, and f is a given function.

The WRM begins by expressing the solution u of (4) as an approximation through a finite sum

$$\mathrm{u}(\mathrm{x}) \approx \mathrm{u}_N(\mathrm{x}) = \sum_{i=0}^{N} \omega_i \phi_i(\mathrm{x}), \tag{5}$$

where $\{\phi_i(\mathrm{x})\}$ are the basis functions and $\{\omega_i\}$ are unknown coefficients. Replacing u with u_N in the (4) yields the residual

$$\mathcal{R}_N(\mathrm{x}) = \mathcal{L}\mathrm{u}_N(\mathrm{x}) - \mathrm{f}(\mathrm{x}), \qquad \mathrm{x} \in \Omega. \tag{6}$$

The unknown coefficients can be determined using either the spectral collocation method or spectral methods of Galerkin type, and we have chosen the spectral collocation method, using Hermite functions as the basis functions. Thus, the approximate solution will be in the form

$$\mathrm{u}_N(\mathrm{x}) = \sum_{i=0}^{N} \omega_i \hat{\mathcal{H}}_i(\mathrm{x}), \tag{7}$$

where $\{\hat{\mathcal{H}}_i(\mathrm{x})\}_{i=0}^{N}$ are the Hermite functions. We aim to determine $\mathrm{u}_N(\mathrm{x})$ at the collocation points $\{\mathrm{x}_i\}_{i=0}^{N}$, which are selected as the roots of Hermite polynomial $\mathcal{H}_{N+1}(\mathrm{x})$.

2.3 Fractional calculus

Fractional calculus is an extension of classical calculus that generalizes the concepts of differentiation and integration to non-integer orders. This advanced mathematical framework provides a powerful tool for modeling complex phenomena that exhibit memory-dependent behavior and inherit past system states, or nonlocal interactions, which are beyond the scope of traditional calculus. Fractional calculus is particularly effective in capturing the dynamics of various phenomena, such as anomalous diffusion, viscoelastic behavior, and power-law scaling, commonly observed in natural and engineered systems. Its versatility has established its significance across a broad range of disciplines, including physics, biology, engineering, and finance, where systems often exhibit long-term dependencies or intricate scaling behaviors. The growing prominence of fractional calculus underscores its critical role in addressing the complexities of modern scientific and engineering challenges.

Three popular definitions of fractional calculus were introduced by Grünwald-Letnikov (GL), Riemann-Liouville (RL), and Caputo. The left and right GL fractional derivatives of order α for a function $\mathrm{u}(\mathrm{x},\mathrm{t})$ with respect to t are defined as follows, respectively:

$${}_{a}^{GL}D_{\mathrm{t}}^{\alpha}\mathrm{u}(\mathrm{x},\mathrm{t}) = \lim_{\xi\to 0} \frac{1}{\xi^{\alpha}} \sum_{\kappa=0}^{\left[\frac{\mathrm{t}-a}{\xi}\right]} (-1)^{\kappa} \binom{\alpha}{\kappa} \mathrm{u}(\mathrm{x},\mathrm{t}-\kappa\xi), \tag{8}$$

$${}_{\mathrm{t}}^{GL}D_{b}^{\alpha}\mathrm{u}(\mathrm{x},\mathrm{t}) = \lim_{\xi\to 0} \frac{1}{\xi^{\alpha}} \sum_{\kappa=0}^{\left[\frac{b-\mathrm{t}}{\xi}\right]} (-1)^{\kappa} \binom{\alpha}{\kappa} \mathrm{u}(\mathrm{x},\mathrm{t}+\kappa\xi), \tag{9}$$

where α is the fractional order of the derivative, a and b are the lower and upper limits of the fractional derivative, respectively, ξ is the time step size, and $\binom{\alpha}{\kappa}$ is the generalized binomial coefficient.

The left and right RL fractional derivatives of order α for a function $u(x,t)$ with respect to t are defined as follows, respectively:

$$ {}^{RL}_{a}D^{\alpha}_{t}u(x,t) = \frac{1}{\Gamma(n-\alpha)}\frac{\partial^n}{\partial t^n}\int_a^t \frac{u(x,s)}{(t-s)^{\alpha-n+1}}ds, \qquad n-1<\alpha<n, \tag{10} $$

$$ {}^{RL}_{t}D^{\alpha}_{b}u(x,t) = \frac{(-1)^n}{\Gamma(n-\alpha)}\frac{\partial^n}{\partial t^n}\int_t^b \frac{u(x,s)}{(s-t)^{\alpha-n+1}}ds, \qquad n-1<\alpha<n, \tag{11} $$

where n is the smallest integer greater than α, and $\Gamma(\cdot)$ is the gamma function, which for a complex number z is defined as

$$ \Gamma(z) = \int_0^{\infty} s^{z-1}e^{-s}ds. $$

The left and right Caputo fractional derivatives of order α for a function $u(x,t)$ with respect to t are defined as follows, respectively:

$$ {}^{C}_{a}D^{\alpha}_{t}u(x,t) = \frac{1}{\Gamma(n-\alpha)}\int_a^t \frac{\partial^n u(x,s)}{\partial s^n}\frac{1}{(t-s)^{\alpha-n+1}}ds, \qquad n-1<\alpha<n, \tag{12} $$

$$ {}^{C}_{t}D^{\alpha}_{b}u(x,t) = \frac{(-1)^n}{\Gamma(n-\alpha)}\int_t^b \frac{\partial^n u(x,s)}{\partial s^n}\frac{1}{(s-t)^{\alpha-n+1}}ds, \qquad n-1<\alpha<n. \tag{13} $$

2.4 Least squares support vector regression

The LS-SVR is a variation of SVR designed to simplify solving regression problems. In LS-SVR, the traditional hinge loss function used in SVR is replaced by a squared loss function, and the inequality constraints are substituted with equality constraints. These adjustments allow LS-SVR to adopt a least-squares approach, making the optimization process more straightforward. As a result, it transforms the original quadratic programming problem into a set of linear equations, which reduces computational complexity while retaining strong generalization capabilities. The LS-SVR is particularly beneficial for handling large datasets or when high computational efficiency is critical.

A significant advantage of LS-SVR lies in its ability to reduce computational complexity in comparison with standard SVR. By solving linear equations instead of complex quadratic problems, LS-SVR scales efficiently to larger datasets while delivering precise regression performance. Additionally, LS-SVR utilizes the kernel trick, similar to SVR, enabling it to capture nonlinear relationships through kernel functions like RBFs or polynomial kernels. This versatility makes LS-SVR a popular choice in fields such as finance, engineering, and machine learning, where accurate regression models are required for tasks such as prediction, time series forecasting, and function approximation.

Consider a given dataset $\{(x_i, y_i)\}_{i=0}^{N}$. The LS-SVR model based on the works of Suykens et al. [52], aims to solve the following optimization problem:

$$ \min_{\mathbf{w},b,\mathbf{e}} \ \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{\zeta}{2}\mathbf{e}^T\mathbf{e}, \tag{14} $$

$$\text{s.t. } \mathrm{y}_j = \mathbf{w}^T \phi(\mathrm{x}_j) + \mathfrak{b} + \mathrm{e}_j, \qquad j = 0, 1, \cdots, N,$$

where $\mathrm{x}_j \in \mathbb{R}^{\kappa}$ such that κ is the dimension of the feature space, $\phi(\cdot) : \mathbb{R}^{\kappa} \to \mathbb{R}^{\kappa_h}$ denotes the feature mapping that transforms input data into a higher-dimensional representation (in this case, we should have $\kappa_h = \kappa$), $\mathbf{w} \in \mathbb{R}^{\kappa_h}$ is the weight vector, $\zeta \in \mathbb{R}^+$ is a regularization parameter related to Tikhonov regularization that controls the trade-off between the training error and the model complexity, $\mathfrak{b} \in \mathbb{R}$ is the bias term, and e_j are the error terms.

Theorem 1. *The solution to problem (14) is uniquely determined by the following linear system*

$$\begin{bmatrix} 0 & \mathbf{1}_{N+1}^T \\ \mathbf{1}_{N+1} & \Phi + \frac{1}{\zeta}\mathbf{I}_{N+1} \end{bmatrix} \begin{bmatrix} \mathfrak{b} \\ \beta \end{bmatrix} = \begin{bmatrix} 0 \\ \mathrm{y} \end{bmatrix}, \tag{15}$$

where $\Phi \in \mathbb{R}^{(N+1)\times(N+1)}$ *is the positive definite kernel such that* $\Phi_{ij} = \phi(\mathrm{x}_i)^T \phi(\mathrm{x}_j)$, $\mathbf{1}_{N+1} = [1, \cdots, 1]^T \in \mathbb{R}^{N+1}$, $\mathbf{I}_{N+1}$ *is the identity matrix of size* $(N+1)\times(N+1)$, $\beta = [\beta_0, \beta_1, \cdots, \beta_N]^T$, *and* $\mathrm{y} = [\mathrm{y}_0, \mathrm{y}_1, \cdots, \mathrm{y}_N]^T$.

Proof. See [52]. □

3 Time-discrete formulation of time-fractional differential equations

The time-fractional diffusion equation (TFDE) is a specialized form of time-fractional differential equation that models diffusion processes with non-standard, typically slower, diffusion rates compared to those predicted by classical models. It is especially useful in describing anomalous diffusion, which occurs in fields such as geophysics, porous media flow, and biological systems. In this equation, the time derivative is of fractional order, allowing it to capture the memory effects that influence particle movement within a medium. This approach provides a more accurate representation of diffusion in complex systems where classical diffusion equations fail to account for the underlying dynamics.

Consider the TFDE of order $0 < \alpha < 1$ as follows:

$${}_0^C D_{\mathrm{t}}^{\alpha} \mathrm{u}(\mathrm{x}, \mathrm{t}) = \gamma \mathrm{u}_{\mathrm{xx}}(\mathrm{x}, \mathrm{t}) + \mathrm{f}(\mathrm{x}, \mathrm{t}), \qquad \mathrm{x} \in \Omega, \ \mathrm{t} \in (0, \mathcal{T}], \tag{16}$$

where $\mathrm{f}(\mathrm{x}, \mathrm{t})$ is a known function.

The weighted and shifted GL approximation, introduced by Tian et al. [55], is an advanced numerical method for solving fractional derivatives, particularly in fractional differential equations. It improves upon the traditional GL approximation by incorporating a weighting scheme and shifting the grid points to enhance both accuracy and stability.

Theorem 2. *Suppose* M *is the number of time steps such that* $1 \le \eta \le M$ *and* ξ *is the time step size. The weighted and shifted GL difference operator for the time-fractional derivative of* $\upsilon(\mathrm{x}, \mathrm{t})$ *is proposed by*

$${}_0^C D_{\mathrm{t}}^{\alpha} \upsilon(\mathrm{x}, \mathrm{t}_\eta) = \frac{1}{\xi^{\alpha}} \sum_{\kappa=0}^{\eta} \omega_{\kappa}^{(\alpha)} \upsilon(\mathrm{x}, \mathrm{t}_{\eta-\kappa}) + \mathcal{O}(\xi^2), \tag{17}$$

in which

$$h_0^{(\alpha)} = 1,$$

$$h_{\kappa}^{(\alpha)} = \left(1 - \frac{\alpha+1}{\kappa}\right) h_{\kappa-1}^{(\alpha)}, \quad \kappa \geq 1,$$

and

$$\omega_0^{(\alpha)} = \left(1 + \frac{\alpha}{2}\right) h_0^{(\alpha)},$$
$$\omega_{\kappa}^{(\alpha)} = \left(1 + \frac{\alpha}{2}\right) h_{\kappa}^{(\alpha)} - \frac{\alpha}{2} h_{\kappa-1}^{(\alpha)}, \quad \kappa \geq 1.$$

Proof. See [55]. □

By using the second order approximation (17) for the time-fractional derivative term in equation (16), we have

$${}_0^C D_t^{\alpha} u(x, t_{\eta}) = \frac{1}{\xi^{\alpha}} \sum_{\kappa=0}^{\eta} \omega_{\kappa}^{(\alpha)} u(x, t_{\eta-\kappa}) + \mathcal{O}(\xi^2),$$

therefor

$${}_0^C D_t^{\alpha} u(x, t_{\eta}) \approx \frac{1}{\xi^{\alpha}} \sum_{\kappa=0}^{\eta} \omega_{\kappa}^{(\alpha)} u(x, t_{\eta-\kappa}). \tag{18}$$

Now, by replacing the approximation (18) in TFDE (16), we have

$$\frac{1}{\xi^{\alpha}} \sum_{\kappa=0}^{\eta} \omega_{\kappa}^{(\alpha)} u(x, t_{\eta-\kappa}) = \gamma u_{xx}(x, t_{\eta}) + f(x, t_{\eta}).$$

With some simplifications, we further have

$$\frac{\omega_0^{(\alpha)}}{\xi^{\alpha}} u^{\eta}(x) - \gamma u_{xx}^{\eta}(x) = f^{\eta}(x) - \frac{1}{\xi^{\alpha}} \sum_{\kappa=1}^{\eta} \omega_{\kappa}^{(\alpha)} u^{\eta-\kappa}(x).$$

By defining

$$\psi^{\eta}(x) = f^{\eta}(x) - \frac{1}{\xi^{\alpha}} \sum_{\kappa=1}^{\eta} \omega_{\kappa}^{(\alpha)} u^{\eta-\kappa}(x),$$

we have time-discrete scheme as follows:

$$\frac{\omega_0^{(\alpha)}}{\xi^{\alpha}} u^{\eta}(x) - \gamma u_{xx}^{\eta}(x) = \psi^{\eta}(x). \tag{19}$$

This can be represented in differential operator form as follows:

$$\mathcal{L} u^{\eta}(x) = \psi^{\eta}(x). \tag{20}$$

4 Hermite collocation LS-SVR for the numerical solutions of time-discrete scheme

Consider a differential equation at time step η in the resulting operator form as follows:

$$\mathcal{L}\mathrm{u}^{\eta}(\mathrm{x}) = \psi^{\eta}(\mathrm{x}).$$

According to [32], we can consider the optimization problem as follows:

$$\begin{aligned} &\min_{\mathbf{w},\mathbf{e}} \ \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{\zeta}{2}\mathbf{e}^T\mathbf{e}, \\ &\text{s.t.} \ \ \mathcal{L}\mathrm{u}^{\eta}(\mathrm{x}_j) - \psi^{\eta}(\mathrm{x}_j) = \mathrm{e}_j^{\eta}, \qquad j = 0,1,\cdots,N, \end{aligned} \tag{21}$$

where the constraints are written as

$$\sum_{i=0}^{N} \mathrm{w}_i^{\eta}\mathcal{L}\phi_i^{\eta}(\mathrm{x}_j) - \psi^{\eta}(\mathrm{x}_j) = \mathrm{e}_j^{\eta}, \qquad j = 0,1,\cdots,N,$$

where $\{\phi_i^{\eta}(\mathrm{x})\}_{i=0}^{N}$ are the Hermite functions at time step η, and $\{\mathrm{x}_j\}_{j=0}^{N}$ are the collocation points given as the roots of Hermite polynomial $\mathcal{H}_{N+1}(\mathrm{x})$.

Now, consider the Lagrangian function of constrained problem (21) as follows:

$$\begin{aligned} \mathscr{L}(\mathbf{w},\beta,\mathbf{e}) = &\frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{\zeta}{2}\mathbf{e}^T\mathbf{e} \\ &- \sum_{j=0}^{N}\beta_j^{\eta}\left(\sum_{i=0}^{N}\mathrm{w}_i^{\eta}\mathcal{L}\phi_i^{\eta}(\mathrm{x}_j) - \psi^{\eta}(\mathrm{x}_j) - \mathrm{e}_j^{\eta}\right), \end{aligned} \tag{22}$$

where $\{\beta_j^{\eta}\}_{j=0}^{N}$ are Lagrange multipliers at time step η. According to the Karush-Kuhn-Tucker conditions, compute the partial derivatives of (22) as follows:

$$\frac{\partial\mathscr{L}}{\partial\mathrm{w}_{\kappa}^{\eta}} = 0 \Longrightarrow \mathrm{w}_{\kappa}^{\eta} - \sum_{j=0}^{N}\beta_j\mathcal{L}\phi_{\kappa}^{\eta}(\mathrm{x}_j) = 0, \tag{23}$$

$$\frac{\partial\mathscr{L}}{\partial\mathrm{e}_{\kappa}^{\eta}} = 0 \longrightarrow \zeta\mathrm{e}_{\kappa}^{\eta} + \beta_{\kappa}^{\eta} = 0 \Longrightarrow \mathrm{e}_{\kappa}^{\eta} = -\frac{1}{\zeta}\beta_{\kappa}^{\eta}, \tag{24}$$

$$\frac{\partial\mathscr{L}}{\partial\beta_{\kappa}^{\eta}} = 0 \Longrightarrow \sum_{i=0}^{N}\mathrm{w}_i^{\eta}\mathcal{L}\phi_i^{\eta}(\mathrm{x}_{\kappa}) - \psi^{\eta}(\mathrm{x}_{\kappa}) - \mathrm{e}_{\kappa}^{\eta} = 0. \tag{25}$$

First, consider (23). Thus we have

$$\mathbf{w}^{\eta} - \mathcal{D}_{\eta}\beta^{\eta} = 0, \tag{26}$$

where

$$\mathcal{D}_\eta = \begin{bmatrix} \mathcal{L}\phi_0^\eta(\mathrm{x}_0) & \mathcal{L}\phi_0^\eta(\mathrm{x}_1) & \mathcal{L}\phi_0^\eta(\mathrm{x}_2) & \cdots & \mathcal{L}\phi_0^\eta(\mathrm{x}_N) \\ \mathcal{L}\phi_1^\eta(\mathrm{x}_0) & \mathcal{L}\phi_1^\eta(\mathrm{x}_1) & \mathcal{L}\phi_1^\eta(\mathrm{x}_2) & \cdots & \mathcal{L}\phi_1^\eta(\mathrm{x}_N) \\ \mathcal{L}\phi_2^\eta(\mathrm{x}_0) & \mathcal{L}\phi_2^\eta(\mathrm{x}_1) & \mathcal{L}\phi_2^\eta(\mathrm{x}_2) & \cdots & \mathcal{L}\phi_2^\eta(\mathrm{x}_N) \\ \vdots & & & \ddots & \vdots \\ \mathcal{L}\phi_N^\eta(\mathrm{x}_0) & \mathcal{L}\phi_N^\eta(\mathrm{x}_1) & \mathcal{L}\phi_N^\eta(\mathrm{x}_2) & \cdots & \mathcal{L}\phi_N^\eta(\mathrm{x}_N) \end{bmatrix},$$

and

$$\mathbf{w}^\eta = \begin{bmatrix} \mathrm{w}_0^\eta \\ \mathrm{w}_1^\eta \\ \mathrm{w}_2^\eta \\ \vdots \\ \mathrm{w}_N^\eta \end{bmatrix}, \quad \beta^\eta = \begin{bmatrix} \beta_0^\eta \\ \beta_1^\eta \\ \beta_2^\eta \\ \vdots \\ \beta_N^\eta \end{bmatrix}.$$

Now, consider (24). So we have

$$\mathbf{e}^\eta = -\frac{1}{\zeta}\beta^\eta, \tag{27}$$

where

$$\mathbf{e}^\eta = \begin{bmatrix} \mathrm{e}_0^\eta \\ \mathrm{e}_1^\eta \\ \mathrm{e}_2^\eta \\ \vdots \\ \mathrm{e}_N^\eta \end{bmatrix}.$$

Finally, consider (25). Then we have

$$\mathcal{D}_\eta^T \mathbf{w}^\eta - \Psi^\eta - \mathbf{e}^\eta = 0, \tag{28}$$

where

$$\Psi^{\eta} = \begin{bmatrix} \psi^{\eta}(\mathrm{x}_0) \\ \psi^{\eta}(\mathrm{x}_1) \\ \psi^{\eta}(\mathrm{x}_2) \\ \vdots \\ \psi^{\eta}(\mathrm{x}_N) \end{bmatrix}.$$

Now, by combining the equations (26), (27) and (28), we have the following linear system

$$\left(\mathcal{D}_{\eta}^{T} + \frac{1}{\zeta}\mathcal{D}_{\eta}^{-1}\right)\mathbf{w}^{\eta} = \Psi^{\eta}. \tag{29}$$

Finally, using the obtained vector $\mathbf{w}^{\eta}$, we can obtain the unknown $\mathrm{u}_N^{\eta}(\mathrm{x})$ at collocation points $\{\mathrm{x}_\kappa\}_{\kappa=0}^{N}$ using the following equation

$$\mathrm{u}_N^{\eta}(\mathrm{x}_\kappa) = \mathbf{w}^{\eta T}\Phi^{\eta}(\mathrm{x}_\kappa), \tag{30}$$

where

$$\Phi^{\eta}(\mathrm{x}_\kappa) = \begin{bmatrix} \phi_0^{\eta}(\mathrm{x}_\kappa) \\ \phi_1^{\eta}(\mathrm{x}_\kappa) \\ \phi_2^{\eta}(\mathrm{x}_\kappa) \\ \vdots \\ \phi_N^{\eta}(\mathrm{x}_\kappa) \end{bmatrix}.$$

4.1 Algorithm of collocation LS-SVR

1. **Input** the number of training data N, number of time steps M, regularization parameter ζ, and fractional order α.
2. **Compute** the collocation points $\{\mathrm{x}_\kappa\}_{\kappa=0}^{N}$ using the roots of Hermite polynomial $\mathcal{H}_{N+1}(\mathrm{x})$.
3. **Compute** the matrix $\mathcal{D}_{\eta}$ and vector Ψ^{η}.
4. **Solve** the linear system $\left(\mathcal{D}_{\eta}^{T} + \frac{1}{\zeta}\mathcal{D}_{\eta}^{-1}\right)\mathbf{w}^{\eta} = \Psi^{\eta}$.
5. **Compute** the unknown $\mathrm{u}_N^{\eta}(\mathrm{x})$ at collocation points $\{\mathrm{x}_\kappa\}_{\kappa=0}^{N}$.

5 Numerical results

This section presents several test problems to demonstrate the accuracy and computational efficiency of the proposed machine learning algorithm. First, we consider two examples to compare spatial and temporal accuracy. Then, we demonstrate the accuracy using additional examples. In all examples, the collocation points $\{x_\kappa\}_{\kappa=0}^{N}$ being the roots of the Hermite polynomial $\mathcal{H}_{N+1}(x)$, have been mapped to the computational domain Ω. All errors are evaluated using the infinity norm, which represents the maximum absolute error over the entire domain. The algorithms were implemented in MATLAB and executed on a notebook computer with a 2.60 GHz CPU and 16 GB of RAM.

Example 1. Consider the TFDE (16) with the parameters $0 < \alpha < 1$ and $\gamma = 1$, where the exact solution is

$$u(x,t) = t^4 \sin(2\pi x).$$

The associated source function $f(x,t)$ is defined as follows:

$$f(x,t) = \left(\frac{\Gamma(5)}{\Gamma(5-\alpha)} t^{4-\alpha} + 4\pi^2 t^4 \right) \sin(2\pi x),$$

where $x \in [0,1]$ and $t \in (0,\mathcal{T}]$.

We compare the collocation LS-SVR method with the compact finite difference method (CFDM) [13] to evaluate the spatial accuracy for various fractional orders α. This comparison involves selecting different values of N while keeping M fixed, to approximate $\{u_N^\eta(x_\kappa); 0 \le \kappa \le N, 1 \le \eta \le M\}$ at the final time $\mathcal{T} = 1$. For this purpose, we choose $N = 2^\ell$ for $\ell = 4,5,6$ and $M = 2^{11}$. For the collocation LS-SVR method, the regularization parameter is set to $\zeta = 10^{-5}$. To assess the spatial accuracy, the numerical errors for different numbers of collocation points are provided in Table 1 for $\alpha = 0.3$, Table 2 for $\alpha = 0.5$, and Table 3 for $\alpha = 0.7$.

Table 1: The numerical errors of the collocation LS-SVR method and CFDM for fractional order α = 0.3 in Example 1

N	collocation LS-SVR	CFDM
16	$1.8176e-04$	$9.5917e-05$
32	$3.2701e-06$	$5.9674e-06$
64	$6.0087e-07$	$3.7257e-07$

Table 2: The numerical errors of the collocation LS-SVR method and CFDM for fractional order α = 0.5 in Example 1

N	collocation LS-SVR	CFDM
16	$1.4571e-04$	$9.4733e-05$
32	$1.4813e-06$	$5.8940e-06$
64	$5.9454e-07$	$3.6834e-07$

Table 3: The numerical errors of the collocation LS-SVR method and CFDM for fractional order α = 0.7 in Example 1

N	collocation LS-SVR	CFDM
16	$1.1006e-04$	$9.3250e-05$
32	$8.7428e-07$	$5.8041e-06$
64	$3.6545e-07$	$3.6505e-07$

According to the results presented in Tables 1-3, the collocation LS-SVR method exhibits good spatial accuracy and convergence behavior. In comparison with the CFDM, the proposed method achieves smaller or comparable errors using the same number of spatial points across various fractional orders α. This superior performance confirms the effectiveness of combining spectral methods with LS-SVR for the spatial discretization of TFDEs.

Example 2. Consider the TFDE (16) with the parameters $0 < \alpha < 1$ and $\gamma = 1$, where the exact solution is

$$u(x,t) = t^2 \sin(\pi x).$$

The corresponding source function $f(x,t)$ is defined as follows:

$$f(x,t) = \left(\frac{\Gamma(3)}{\Gamma(3-\alpha)} t^{2-\alpha} + \pi^2 t^2 \right) \sin(\pi x),$$

where $x \in [0,1]$ and $t \in (0,\mathcal{T}]$.

We compare the collocation LS-SVR method with the FDM on a uniform mesh [62] to evaluate temporal accuracy for different fractional orders α. In this comparison, a fixed value of N is selected such that the spatial discretization error remains negligible in comparison to the temporal error, and various values of M to approximate $\{u_N^{\eta}(x_\kappa); 0 \le \kappa \le N, 1 \le \eta \le M\}$ at the final time $\mathcal{T} = 1$. For this purpose, we set $N = 100$ and $M = 10, 20, 40, 80$. The regularization parameter for the collocation LS-SVR method is set to $\zeta = 10^{-5}$. To determine temporal accuracy, the numerical errors for different numbers of time steps are provided in Table 4 for $\alpha = 0.3$, Table 5 for $\alpha = 0.5$, and Table 6 for $\alpha = 0.7$.

Table 4: The numerical errors of the collocation LS-SVR method and FDM for fractional order α = 0.25 in Example 2

M	collocation LS-SVR	FDM
10	$9.3013e-05$	$3.5481e-04$
20	$2.2570e-05$	$1.1159e-04$
40	$5.2912e-06$	$3.4703e-05$
80	$1.1376e-06$	$1.0700e-05$

Table 5: The numerical errors of the collocation LS-SVR method and FDM for fractional order $\alpha = 0.5$ in Example 2

M	collocation LS-SVR	FDM
10	$1.5808e-04$	$1.3218e-03$
20	$3.8378e-05$	$4.7736e-04$
40	$9.1865e-06$	$1.7126e-04$
80	$4.9408e-05$	$6.1171e-05$

Table 6: The numerical errors of the collocation LS-SVR method and FDM for fractional order $\alpha = 0.75$ in Example 2

M	collocation LS-SVR	FDM
10	$1.4752e-04$	$3.8471e-03$
20	$3.5669e-05$	$1.6274e-03$
40	$8.4410e-06$	$6.8665e-04$
80	$3.0657e-06$	$2.8929e-04$

Based on the results presented in Tables 4-6, the collocation LS-SVR method exhibits strong temporal accuracy and convergence performance. Compared to the FDM, the proposed method achieves smaller errors for the same number of time steps. This superior performance highlights the effectiveness of combining the weighted and shifted GL approximation with the LS-SVR approach for time-fractional problems.

Example 3. Consider the TFDE (16) with the parameters $0 < \alpha < 1$ and $\gamma = 1$, where the exact solution is

$$u(x,t) = t^{\mathfrak{p}} \cos(\pi x).$$

The associated source function $f(x,t)$ is defined as follows:

$$f(x,t) = \left(\frac{\Gamma(\mathfrak{p}+1)}{\Gamma(\mathfrak{p}+1-\alpha)} t^{\mathfrak{p}-\alpha} + \pi^2 t^{\mathfrak{p}} \right) \cos(\pi x),$$

where $x \in [0,1]$ and $t \in (0,\mathcal{T}]$.

To analyze the spatial accuracy for different fractional order α, we vary N while keeping M fixed to approximate $\{u_N^{\eta}(x_\kappa); 0 \le \kappa \le N, 1 \le \eta \le M\}$ at the final time $\mathcal{T} = 1$. For this purpose, we set $N = 5, 10, 20, 40$ and $M = 2^{10}$. The regularization parameter is chosen as $\zeta = 10^{-5}$. The numerical errors and CPU times for different numbers of collocation points and various values of $\mathfrak{p}$ are presented in Table 7 for $\alpha = 0.25$, Table 8 for $\alpha = 0.5$, Table 9 for $\alpha = 0.75$. Figure 1 illustrates numerical errors obtained based on the different fractional orders α and parameter $\mathfrak{p} = 4$, for various numbers of nodes N.

Table 7: The numerical errors and CPU times of the collocation LS-SVR method for fractional order $\alpha = 0.25$ in Example 3

	$\mathfrak{p}=3$		$\mathfrak{p}=4$		$\mathfrak{p}=5$	
N	E_∞	CPU-T(s)	E_∞	CPU-T(s)	E_∞	CPU-T(s)
5	$5.4500e-04$	20.07243	$5.4227e-04$	20.00604	$5.4001e-04$	19.52457
10	$2.2991e-05$	28.21557	$2.2946e-05$	28.25700	$2.2904e-05$	27.22121
20	$5.5092e-06$	55.86149	$5.4989e-06$	56.54565	$5.4869e-06$	58.08762
40	$8.5667e-07$	194.13622	$8.5294e-07$	195.93890	$8.4744e-07$	203.17477

Table 8: The numerical errors and CPU times of the collocation LS-SVR method for fractional order $\alpha = 0.5$ in Example 3

	$\mathfrak{p}=3$		$\mathfrak{p}=4$		$\mathfrak{p}=5$	
N	E_∞	CPU-T(s)	E_∞	CPU-T(s)	E_∞	CPU-T(s)
5	$5.0012e-04$	19.61594	$4.9287e-04$	19.74157	$4.8667e-04$	20.25583
10	$1.6999e-05$	24.29448	$1.6877e-05$	25.31449	$1.6753e-05$	25.67332
20	$4.2587e-06$	49.40230	$4.2324e-06$	49.54202	$4.1976e-06$	47.81303
40	$1.0689e-06$	166.16599	$1.0568e-06$	167.53989	$1.0385e-06$	165.33069

Table 9: The numerical errors and CPU times of the collocation LS-SVR method for fractional order $\alpha = 0.75$ for Example 3

	$\mathfrak{p}=3$		$\mathfrak{p}=4$		$\mathfrak{p}=5$	
N	E_∞	CPU-T(s)	E_∞	CPU-T(s)	E_∞	CPU-T(s)
5	$4.6797e-04$	21.46464	$4.5408e-04$	20.69291	$4.4200e-04$	19.74179
10	$8.7988e-06$	27.04170	$8.6116e-06$	28.78288	$8.4006e-06$	28.58350
20	$3.2333e-06$	56.50055	$3.1816e-06$	57.80940	$3.1045e-06$	58.34600
40	$8.5895e-07$	199.88857	$8.3353e-07$	201.42734	$7.9103e-07$	213.78422

The numerical results presented in Tables 7-9 demonstrate that the collocation LS-SVR method achieves excellent spatial accuracy using relatively few collocation points. As the number of collocation points increases, the maximum absolute errors decrease rapidly, which confirms the spectral accuracy property of the Hermite function approximation. This significant error reduction with modest increases in the number of basis functions highlights the computational efficiency of the proposed method. Furthermore, the consistency of the results across different values of the fractional order α and the parameter $\mathfrak{p}$ illustrates the robustness and reliability of the collocation LS-SVR approach for solving TFDEs. Additionally, the CPU times reported in these tables reveal that the proposed method exhibits a favorable balance between accuracy and computational cost, as the increase in computational effort remains moderate with respect to the number of collocation points. This further confirms the practical efficiency of the collocation LS-SVR approach for problems requiring high-precision solutions.

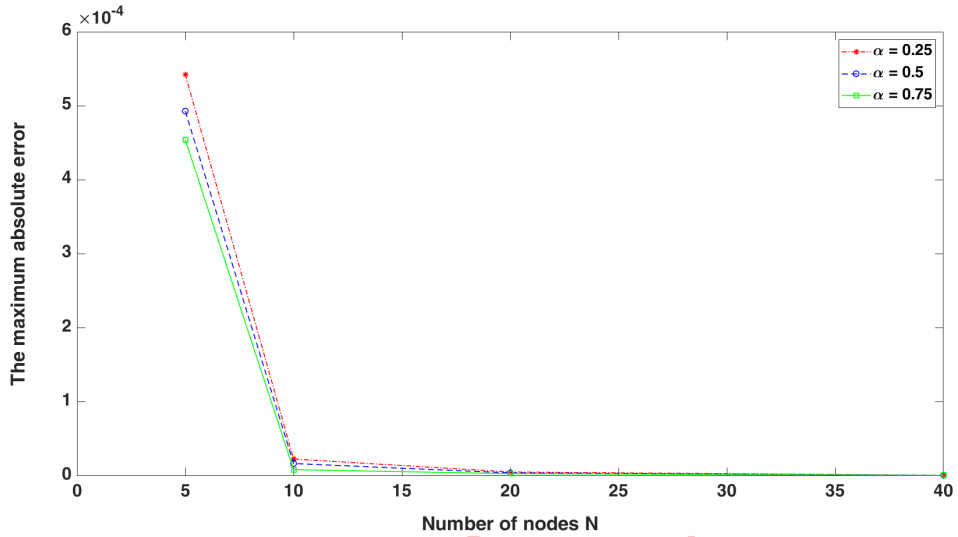

Figure 1: The maximum absolute errors of the collocation LS-SVR method across various fractional orders α and parameter $\mathfrak{p} = 4$

Example 4. Consider the TFDE (16) with the parameters $0 < \alpha < 1$ and $\gamma = 1$, where the exact solution is

$$u(x,t) = t^{\mathfrak{p}} \cos(2\pi x).$$

The corresponding source function $f(x,t)$ is defined as follows:

$$f(x,t) = \left(\frac{\Gamma(\mathfrak{p}+1)}{\Gamma(\mathfrak{p}+1-\alpha)} t^{\mathfrak{p}-\alpha} + 4\pi^2 t^{\mathfrak{p}} \right) \cos(2\pi x),$$

where $x \in [0,1]$ and $t \in (0,\mathcal{T}]$.

To evaluate the temporal accuracy for various fractional orders α, we fix N at a sufficiently large value so that the spatial error is negligible compared to the temporal error and vary M to approximate $\{u_N^{\eta}(x_\kappa); 0 \le \kappa \le N, 1 \le \eta \le M\}$ at the final time $\mathcal{T} = 1$. For this purpose, we set $M = 5, 10, 20, 40, 80$ with the regularization parameter $\zeta = 10^{-5}$. The numerical errors and temporal convergence orders for different numbers of time steps and various values of $\mathfrak{p}$ are presented in Table 10 for $\alpha = 0.25$ and $N = 100$, Table 11 for $\alpha = 0.5$ and $N = 150$, and Table 12 for $\alpha = 0.75$ and $N = 100$. Figure 2 illustrates numerical errors obtained based on the different fractional orders α and parameter $\mathfrak{p} = 4$, for various numbers of time steps M.

Tables 10-12 indicate that the temporal convergence order approaches 2 for various fractional orders α, which confirms the theoretical result established in Theorem 2. Specifically, as the number of time steps M increases, the numerical errors decrease at a rate consistent with second-order accuracy. This behavior is observed across different values of the fractional order α and the parameter $\mathfrak{p}$, demonstrating the robustness and reliability of the proposed method. Furthermore, the computed convergence orders presented in these tables are in excellent agreement with the theoretical convergence rate of the weighted and shifted GL approximation, thereby validating the accuracy of the time discretization scheme employed in this study.

Table 10: The numerical errors and temporal convergence orders of the collocation LS-SVR method for fractional order $\alpha = 0.25$ in Example 4

	$\mathfrak{p}=3$		$\mathfrak{p}=4$		$\mathfrak{p}=5$	
M	E_∞	C-Order	E_∞	C-Order	E_∞	C-Order
5	$6.7137e-04$	–	$1.3689e-03$	–	$2.2385e-03$	–
10	$1.7507e-04$	1.9392	$3.7534e-04$	1.8667	$6.4375e-04$	1.7979
20	$4.4362e-05$	1.9805	$9.7730e-05$	1.9413	$1.7194e-04$	1.9046
40	$1.1583e-05$	1.9373	$2.5389e-05$	1.9446	$4.4917e-05$	1.9365
80	$3.1507e-06$	1.8783	$6.6552e-06$	1.9317	$1.1658e-05$	1.9459

Table 11: The numerical errors and temporal convergence orders of the collocation LS-SVR method for fractional order $\alpha = 0.5$ in Example 4

	$\mathfrak{p}=3$		$\mathfrak{p}=4$		$\mathfrak{p}=5$	
M	E_∞	C-Order	E_∞	C-Order	E_∞	C-Order
5	$1.6109e-03$	–	$3.6734e-03$	–	$6.3883e-03$	–
10	$4.1693e-04$	1.9500	$1.0117e-03$	1.8604	$1.8622e-03$	1.7784
20	$1.0641e-04$	1.9701	$2.6542e-04$	1.9304	$5.0249e-04$	1.8898
40	$2.6171e-05$	2.0236	$6.7223e-05$	1.9812	$1.2974e-04$	1.9535
80	$6.9500e-06$	1.9129	$1.7373e-05$	1.9521	$3.3412e-05$	1.9572

Table 12: The numerical errors and temporal convergence orders of the collocation LS-SVR method for fractional order $\alpha = 0.75$ in Example 4

	$\mathfrak{p}=3$		$\mathfrak{p}=4$		$\mathfrak{p}=5$	
M	E_∞	C-Order	E_∞	C-Order	E_∞	C-Order
5	$2.7822e-03$	–	$7.2066e-03$	–	$1.3325e-02$	–
10	$7.1055e-04$	1.9692	$1.9880e-03$	1.858	$3.9309e-03$	1.7612
20	$1.7966e-04$	1.9836	$5.2084e-04$	1.9324	$1.0655e-03$	1.8833
40	$4.5345e-05$	1.9863	$1.3339e-04$	1.9652	$2.7738e-04$	1.9416
80	$1.1481e-05$	1.9816	$3.3807e-05$	1.9803	$7.0789e-05$	1.9703

6 Conclusions

In this study, a machine learning-based numerical scheme was introduced for solving time-fractional differential equations. The method employs least squares support vector regression with a collocation approach, utilizing Hermite functions as basis functions and the roots of Hermite polynomials $\mathcal{H}_{N+1}(x)$ as collocation points. The weighted and shifted Grünwald-Letnikov approximation was applied for the discretization of the time-fractional derivative. By considering several benchmark problems with known

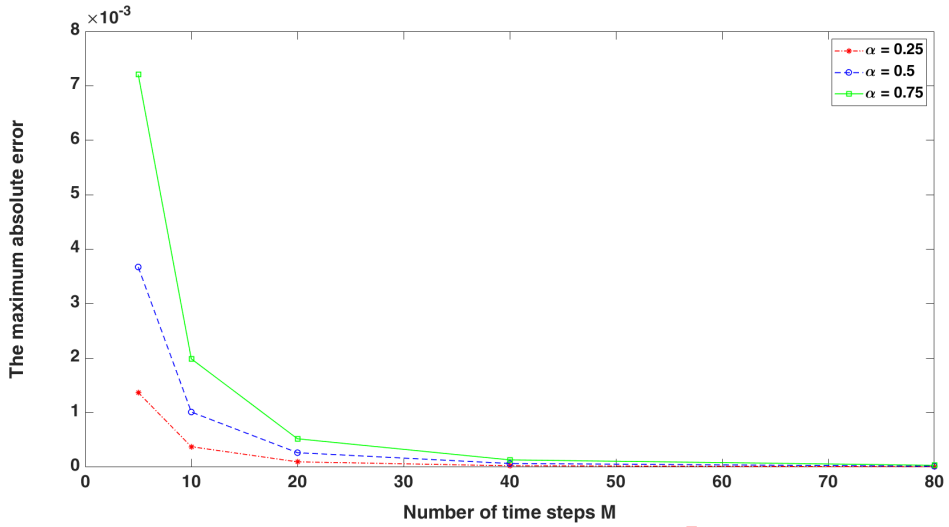

Figure 2: The maximum absolute errors of the collocation LS-SVR method across various fractional orders α and parameter $\mathfrak{p} = 4$

analytical solutions and examining the accuracy and computational efficiency of the proposed method, it is shown that the numerical scheme achieves second-order temporal convergence. This result is consistent with the theoretical order of accuracy established for the time discretization technique. Additionally, the method exhibits good spatial accuracy and computational efficiency across various test problems.

Acknowledgements

The authors would like to express their sincere gratitude to both reviewers for their time and insightful feedback, which significantly enhanced the quality of this paper.

Conflict of Interest

The authors declare no competing financial interests or personal relationships that could have influenced this work.

References

[1] M. Abbaszadeh, A.R. Bagheri Salec, A.S. Jebur, *Application of compact local integrated RBFs technique to solve fourth-order time-fractional diffusion-wave system*, J. Math. Model. **12(3)** (2024) 431–449.

[2] M. Abbaszadeh, M. Dehghan, *Meshless upwind local radial basis function-finite difference technique to simulate the time-fractional distributed-order advection–diffusion equation*, Eng. Comput. **37** (2021) 873–889.

[3] M. Abbaszadeh, M.A. Zaky, A.S. Hendy, M. Dehghan, *Supervised learning and meshless methods for two-dimensional fractional PDEs on irregular domains*, Math. Comput. Simul. **216** (2024) 77–103.

[4] R. Abedian, *WENO schemes with Z-type non-linear weighting procedure for fractional differential equations*, J. Math. Model. **10(4)** (2022) 555–567.

[5] M. Adel, S. Abdi-mazraeh, G. Zaki, S. Irandust-Pakchin, *The method based on quintic B-spline functions for addressing time-fractional advection-dispersion equations*, J. Math. Model. **14(2)** (2025) 533–556.

[6] A. Babaei, *Solving a time-fractional inverse heat conduction problem with an unknown nonlinear boundary condition*, J. Math. Model. **7(1)** (2022) 85–106.

[7] D. Baleanu, K. Diethelm, E. Scalas, J.J. Trujillo, *Fractional Calculus: Models and Numerical Methods*, World Sci., 2012.

[8] Z. Barary, A.B. Yazdani Cherati, S. Nemati, *An efficient numerical scheme for solving a general class of fractional differential equations via fractional-order hybrid jacobi functions*, Comm. Nonlinear Sci. Numer. Simul. **128** (2024) 107599.

[9] N. Biranvand, A.M. Salari, *Energy estimate for impulsive fractional advection dispersion equations in anomalous diffusions*, J. Nonlinear Funct. Anal **2018** (2018) 1–17.

[10] N. Biranvand, A. Ebrahimijahan, *Numerical study of the multi-dimensional Galilei invariant fractional advection–diffusion equation using direct mesh-less local Petrov–Galerkin method*, Eng. Anal. Bound. Elem. **167** (2024) 105910.

[11] M. Bolhassani, H. Dana Mazraeh, K. Parand, *A new method based on least-squares support vector regression for solving optimal control problems*, Kybernetika **60** (2024) 513–534.

[12] C. Canuto, M. Y. Hussaini, A. Quarteroni, T. A. Zang, *Spectral Methods*, Springer, 2006.

[13] J. Cao, Z. Wang, Z. Wang, *Stability and convergence analysis for a uniform temporal high accuracy of the time-fractional diffusion equation with 1D and 2D spatial compact finite difference method*, AIMS Math. **9** (2024) 14697–14730.

[14] C. Cortes, *Support-vector networks*, Mach. Learn. **20** (1995) 273–297.

[15] H. Dehestani, Y. Ordokhani, M. Razzaghi, *Ritz-least squares support vector regression technique for the system of fractional Fredholm-Volterra integro-differential equations*, J. Appl. Math. Comput. **71** (2025) 3477–3508.

[16] M. Dehghan, M. Abbaszadeh, A. Mohebbi, *Analysis of two methods based on Galerkin weak form for fractional diffusion-wave: meshless interpolating element free Galerkin (IEFG) and finite element methods*, Eng. Anal. Bound. Elem. **64** (2016) 205–221.

[17] K. Diethelm, N.J. Ford, *Analysis of fractional differential equations*, J. Math. Anal. Appl. **265** (2002) 229–248.

[18] B.A. Finlayson, *The Method of Weighted Residuals and Variational Principles*, SIAM, 2013.

[19] A. Ghoreyshi, M. Abbaszadeh, M.A. Zaky, M. Dehghan, *An accurate and robust numerical method for solving distributed-order space–time fractional PDEs*, Z. Angew. Math. Phys. **76** (2025) 242.

[20] D. Gottlieb, S. A. Orszag, *Numerical Analysis of Spectral Methods: Theory and Applications*, SIAM, 1977.

[21] S. Irandoust-Pakchin, M. Javidi, H. Kheiri, *Analytical solutions for the fractional nonlinear cable equation using a modified homotopy perturbation and separation of variables methods*, Comput. Math. Math. Phys. **56** (2016) 116–131.

[22] S. Irandoust-Pakchin, S. Abdi-Mazraeh, M. Adel, *Application of flatlet oblique multiwavelets to solve the fractional stochastic integro-differential equation using Galerkin method*, Math. Methods Appl. Sci. **47**, (2024) 8342–8365.

[23] S. Irandoust-Pakchin, *Exact solutions for some of the fractional differential equations by using modification of He's variational iteration method*, Math. Sci. **5** (2011) 51–60.

[24] S. Irandoust-Pakchin, S. Abdi-Mazraeh, I. Fahimi-Khalilabad, *Higher order class of finite difference method for time-fractional Liouville-Caputo and space-Riesz fractional diffusion equation*, Filomat **38** (2024) 505–521.

[25] S. Irandoust-Pakchin, M. Lakestani, H. Kheiri, *Numerical approach for solving a class of nonlinear fractional differential equation*, Bull. Iran. Math. Soc. **42** (2016) 1107–1126.

[26] H. Jafari, B.F. Malidareh, V.R. Hosseini, *Collocation discrete least squares meshless method for solving nonlinear multi-term time fractional differential equations*, Eng. Anal. Bound. Elem. **158** (2024) 107–120.

[27] M.M. Khader, K.M. Saad, Z. Hammouch, D. Baleanu, *A spectral collocation method for solving fractional KdV and KdV-Burgers equations with non-singular kernel derivatives*, Appl. Numer. Math. **161** (2021) 137–146.

[28] A. Kilbas, H.M. Srivastava, J.J. Trujillo, *Theory and Applications of Fractional Differential Equations*, Elsevier, 2006.

[29] C. Li, F. Zeng, *Numerical Methods for Fractional Calculus*, CRC Press, 2015.

[30] S. Li, H. Ding, *Numerical analysis of two-dimensional time-fractional Allen-Cahn equation on a new non-uniform mesh construction strategy*, J. Sci. Comput. **104** (2025) 81.

[31] H.R. Marzban, A. Nezami, *Analysis of nonlinear fractional optimal control systems described by delay Volterra–Fredholm integral equations via a new spectral collocation method*, Chaos Solit. Fractals **162** (2022) 112499.

[32] S. Mehrkanoon, T. Falck, J. A.K. Suykens, *Approximate solutions to ordinary differential equations using least squares support vector machines*, IEEE Trans. Neural Netw. Learn. Syst. **23** (2012) 1356–1367.

[33] A. Mohammadi, A. Tari, *A new approach to numerical solution of the time-fractional KdV-Burgers equations using least squares support vector regression*, J. Math. Model., **12(4)** (2024) 583–602.

[34] H. Mohammadi-Firouzjaei, H. Adibi, M. Dehghan, *A comparative study on interior penalty discontinuous Galerkin and enriched Galerkin methods for time-fractional Sobolev equation*, Eng. Comput. **38** (2022) 5379–5394.

[35] M. Molavi-Arabshahi, J. Rashidinia, M. Yousefi, *An efficient approach for solving the fractional model of the human T-cell lymphotropic virus I by the spectral method*, J. Math. Model. **11(3)** (2023) 463-477.

[36] N. Najafi, N. Biranvand, *Solving fuzzy impulsive fractional differential equations by reproducing kernel Hilbert space method*, Internat. J. Math. Model. Comput. **10** (2020) 37–56.

[37] O. Nikan, J. Rashidinia, H. Jafari, *Numerically pricing American and European options using a time fractional Black–Scholes model in financial decision-making*, Alexandria Eng. J. **112** (2025) 235–245.

[38] A. Pakniyat, K. Parand, M. Jani, *Least squares support vector regression for differential equations on unbounded domains*, Chaos Solit. Fractals **151** (2021) 111232.

[39] K. Parand, A. Aghaei, M. Jani, A. Ghodsi, *A new approach to the numerical solution of Fredholm integral equations using least squares-support vector regression*, Math. Comput. Simul. **180** (2021) 114–128.

[40] K. Parand, M. Hasani, M. Jani, H. Yari, *Numerical simulation of Volterra–Fredholm integral equations using least squares support vector regression*, Comput. Appl. Math. **40** (2021) 1–15.

[41] K. Parand, M. Razzaghi, R. Sahleh, M. Jani, *Least squares support vector regression for solving Volterra integral equations*, Eng. Comput., **38** (2022) 789–796.

[42] X. Peng, W. Qiu, A.S. Hendy, M.A. Zaky, *Temporal second-order fast finite difference/compact difference schemes for time-fractional generalized Burgers' equations*, J. Sci. Comput. **99** (2024) 52.

[43] I. Podlubny, *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*, Elsevier, 1998.

[44] L. Qing, X. Li, *Analysis of a meshless generalized finite difference method for the time-fractional diffusion-wave equation*, Comput. Math. Appl. **172** (2024) 134–151.

[45] M. Ramezani, R. Mokhtari, *Numerical solution of distributed-order fractional diffusion equations using a high-order temporal scheme*, Commun. Appl. Math. Comput. (2025) 1–15.

[46] F.A. Shah, K. Shah, T. Abdeljawad, *Numerical solution of two dimensional time-fractional telegraph equation using Chebyshev spectral collocation method*, Partial Differ. Equ. Appl. Math. **13** (2025) 101129.

[47] J. Shen, T. Tang, L.-L. Wang, *Spectral Methods: Algorithms, Analysis and Applications*, Springer Sci. Bus. Media, 2011.

[48] J. Shi, X. Yang, X. Liu, *A novel fractional physics-informed neural networks method for solving the time-fractional Huxley equation*, Neural Comput. Appl, **36** (2024) 19097–19119.

[49] H. Payandehdoost Masouleh, M. Esmailzadeh, *An efficient numerical method based on cubic B–splines for the time–fractional Black–Scholes European option pricing model*, J. Math. Model., **12(3)** (2024) 405–417.

[50] P.K. Singh, S.S. Ray, *A numerical approach based on Pell polynomial for solving stochastic fractional differential equations*, Numer. Algorithms **97** (2024) 1513–1534.

[51] S.M. Sivalingam, V. Govindaraj, *A novel numerical approach for time-varying impulsive fractional differential equations using theory of functional connections and neural network*, Expert Syst. Appl. **238** (2024) 121750.

[52] J.A K. Suykens, J. De Brabanter, L. Lukas, J. Vandewalle, *Weighted least squares support vector machines: robustness and sparse approximation*, Neurocomput. **48** (2002) 85–105.

[53] J.A K. Suykens, J. Vandewalle, *Least squares support vector machine classifiers*, Neural Proc. Lett. **9** (1999) 293–300.

[54] S. Thakur, H. Mitra, A.M. Ardekani, *Physics-informed neural network-based inverse framework for time-fractional differential equations for rheology*, Biol, **14** (2025) 779.

[55] W.Y. Tian, H. Zhou, W.H. Deng, *A Class of Second Order Difference Approximations for Solving Space Fractional Diffusion Equations*, Math. Comput. **84** (2015) 1703–1727.

[56] L.N. Trefethen, *Spectral Methods in MATLAB*, SIAM, 2000.

[57] F. Wang, Q. Zhao, Z. Chen, C.-M. Fan, *Localized Chebyshev collocation method for solving elliptic partial differential equations in arbitrary 2D domains*, Appl. Math. Comput. **397** (2021) 125903.

[58] J. Wang, X. Shen, Y. Liu, S. Guo, *High-order energy stable algorithm for time-fractional Swift-Hohenberg model on graded meshes*, J. Sci. Comput. **104** (2025) 93.

[59] M. Xie, S.U. Khan, W. Sumelka, A.M. Alamri, S.A. AlQahtani, *Advanced stability analysis of a fractional delay differential system with stochastic phenomena using spectral collocation method*, Sci. Rep. **14** (2024) 12047.

[60] M. Zakaria, A. Moujahid, *On Galerkin spectral element method for solving Riesz fractional diffusion equation based on Legendre polynomials*, J. Math. Model., **13(2)** (2025) 281–301.

[61] T. Zhang, D. Zhang, S. Shi, Z. Guo, *Spectral-fPINNs: spectral method based fractional physics-informed neural networks for solving fractional partial differential equations*, Nonlinear Dyn. **113** (2025) 12565–12588.

[62] Y. Zhang, Z. Sun, H. Liao, *Finite difference methods for the time fractional diffusion equation on non-uniform meshes*, J. Comput. Phys. **265** (2014) 195–210.

[63] T. Zhao, C. Li, D. Li, *Efficient spectral collocation method for fractional differential equation with Caputo-Hadamard derivative*, Fract. Calc. Appl. Anal. **26** (2023) 2903–2927.

[64] Y. Zhou, *Basic Theory of Fractional Differential Equations*, World Sci., 2023.

[65] Y. Zhou, C. Li, M. Stynes, *A fast second-order predictor-corrector method for a nonlinear time-fractional Benjamin-Bona-Mahony-Burgers equation*, Numer. Algorithms **95** (2024) 693–720.