DJS Vol. 45(1) (2022) pp.75-96 ISSN: 1012-5965

Delta Journal of Science
Available online at
https://djs.journals.ekb.eg/

Research Article MATHEMATICS

A New Numerical Mechanism for Solving Two Models of Variable Order Delay Differential Equations

Hoda F. Ahmed[1], Ayman M. Mahmoud[2], Marina B. Melad[2]

[1]Department of Mathematics, Faculty of Science, Minia University, Minia 61519, Egypt

[2]Department of Mathematics, Faculty of Science, New Valley University, El-Khargah 72511, Egypt

Corresponding author: Marina B. Melad e-mail: Marina@Sci.nvu.edu.eg

Received: 15 /10/2022 Accepted: 18/12/2022:

KEY WORDS

Variable order delay differential equations with initial functions; Variable order Volterra delay integro-differential equations; Operational matrix; Shifted fractional Gegenbauer polynomials; The method of steps; Collocation method

ABSTRACT

The current paper offers an effective numerical mechanism for solving two models of variable order (VO) linear/nonlinear delay differential equations; the models represent the variable order delay differential equations (VODDEs) and the variable order Volterra delay integro-differential equations (VO-VDIDES) with initial functions. In the proposed mechanism, the method of steps is used to transfer the VO delay models (VO-DMs) into variable order non delay ones with initial conditions. After a novel operational matrix (OM) of VO derivative of the shifted fractional Gegenbauer polynomials (SFGPs) injunction with the spectral collocation method are utilized to transfer the aforementioned problem into a system of algebraic equations, which is simple to solve. The error estimation of the proposed technique is established through the article. The efficiency and accuracy of the proposed technique are verified by applying it to several numerical delay differential equations with constant or variable delay. The obtained numerical results are compared with other published data in the existing literature to expose the accuracy and efficiency of the proposed technique.

Introduction

Recently, fractional calculus has received a lot of interest of many researchers; it allows us to analyze integration and differentiation of any order that isn't necessarily an integer. The fractional operator can better describe a variety of real-world phenomena than the integer order calculus [19, 1, 20]. The most familiar types of the fractional integrals and derivatives are the constant-order (CO) and VO. According to many researchers in recent years, many orders of differential equations that describe natural phenomena with fractional order can alter across time and space [11]. The VO models have gotten a lot of attentions, due to their appropriateness in simulating a wide range of phenomena across various sectors of science and engineering, such as anomalous diffusion [32], viscoelastic mechanics [13], control systems [21], petroleum engineering [27], and many other branches of physics and engineering [33, 14]. In 2016, Almeida et al. [1] looked into the topic of the population expansion problem by utilizing a fractional differential equation with a constant fractional-order to explain the dynamics. They expanded on previous research in 2018 [2] by considering the order as a function of time. In comparison to the constant order model, their research showed that the VO model is far more effective at estimating global population rise. Delay differential equation (DDE) is differed from the ordinary differential equation in that the derivative at every time is reliant on the solution at previous times. The DDE is also called a difference-differential equation or functional differential equation or an equation with a time lag or a differential equation with deviating arguments. Distinct forms of DDEs are existed, each with a different sort of delay, such as time-dependent delay, neutral delay, constant delay, state-dependent delay, stochastic delay, and so forth. DDEs are used in a variety of fields, including biology, electrical networks, and population dynamics; get back to [30, 10] and their references. Fractional delay models (FDMs) are gaining popularity as a new branch of nonlinear analysis. They provide a more natural foundation for modeling mathematically many real-world phenomena. They have a variety of applications, like electrodynamics, astrophysics, nonlinear dynamical systems, probability theory on algebraic structure [24]. In this area, tremendous progress has been made, with the majority of these publications devoted to the study of FDMs. On the other hand, classical calculus, cannot always provide the most accurate representation of some complicated events, such as those observed in biological systems and medicine, so VO-DDEs have arisen. It is known that the exact solution (ES) for such models is not available, so the main objective of this research article is to present and develop an accurate and effective numerical technique to solve some VO-DDEs. There are few works that appeared in this direction like; Abdelkawy [3] proposed the shifted fractional Jacobi collocation method for solving VO-DDEs; Dehghan [15] proposed numerical approach for solving a class of VO fractional functional boundary value problems (BVPs); Yang et al [31] proposed an algorithm depended on the reproducing kernel splines method for solving VO-DDEs; Li et al., [23] solved the VO-DDEs using Reproducing kernel method; Bhrawy and Zaky [12] used Chebyshev operational matrix to solve delay Dirichlet boundary value problem with a type of VO Caputo fractional derivative; Li and Wu. [22] presented numerical method for solving VO functional BVPs.

The primary goal of this research is to extract an accurate numerical method to solve the following models:

- VO-DDEs

$$D_t^{\nu(t)}y(t) = F(t, y(t), y(t-\tau)), t_0 \leq t \leq t_f, 0 < \nu(t) \leq 1, \qquad (1.1)$$

With the initial function (IF)

$$y(t) = g(t), \qquad t \in [-\tau, 0], \qquad (1.2)$$

Where $\nu(t), g(t)$ and $F(t, y(t), y(t-\tau))$ are given functions which may be linear/nonlinear functions, τ is the delay which may be constant or variable, $D_t^{\nu(t)}$ denotes the VO Caputo derivative and $y(t)$ is unknown function.

- VO-VDIDES

$$D_t^{\nu(t)} y(t) = f_1(t, y(t)) + \int_{t_0}^{t} K(y(\xi-\tau)) d\xi, t_0 \leq t \leq t_f, 0 < \nu(t) \leq 1, \qquad (1.3)$$

With the IF

$$y(t_0) = y_0,$$
$$y(t) = q(t), t \in [-\tau, t_0], \qquad (1.4)$$

Where $\nu(t), q(t), f_1(t, y(t))$ and $K(y(\xi-\tau))$ are linear/non-linear functions, τ is the delay which may be constant or variable, $D_t^{\nu(t)}$ denotes the VO Caputo derivative and $y(t)$ is unknown function.

Spectral techniques are one of the most effective and powerful numerical algorithms for solving differential and integral equations with integer and non-integer order derivatives. Spectral approaches such as pseudo spectral [4], Galerkin [5], and Tau [17] can be produced from a weighted residual method. The spectral collocation method is becoming more widely used, with applications in a variety of fields. It is characterized by exponential rates of convergence, great precision, ease of use, and effectiveness in handling a variety of problems. Besides the spectral methods, operational matrices are commonly utilized for solving these types of equations [17, 5]. The current paper is based on the shifted fractional Gegenbauer polynomials for building the variable-order operational matrix of derivatives. Our choice of SFGPs gets back to several characteristics of these polynomials:

- For a small set of spectral expansion terms, they attain quick convergence rates.
- The Gegenbauer polynomials (GPs) have been proven to be particularly efficient in solving a variety of problems in numerous research [17, 6, 18, 7, 16].
- SFGPs are determined by two parameters α and μ. For the first parameter $\alpha > -0.5$. Each change of this parameter creates a new polynomial like the first kind of shifted fractional Chebyshev polynomials (SFCPs) with the parameter $\alpha = 0,$ the second kind of SFCPs with the parameter $\alpha = 1,$ and the shifted fractional Legendre polynomials (SFLPs) with the parameter $\alpha = \frac{1}{2}.$ The second parameter $\mu > 0,$ plays an important role in dealing with the problems which have singular or non-smooth solutions by suitably chosen of its value.
- The SFGPs' OM techniques have been proven effectiveness in solving many problems. Ahmed and Melad [8] derived a new SFGOM of differentiation and integration in conjunction with the spectral collocation method to solve VO pantograph-delay differential equations and VO pantograph Volterra delay integro-differential equations, El-Gindy *et al.*, [19] used different kinds of SFGOMs of the VO differentiation and integration to solve the VO Fredholm–Volterra integro-differential equations, systems of VO-FV-IDEs and VO Volterra partial integro-differential equations, Ahmed and Melad [9] extracted a novel numerical strategy that depends on SFGPs to solve nonlinear singular Emden-Fowler VO-DMs, EL-Kalaawy *et al.* [18] proposed SFGOM of integration to numerically find the solution of fractional variation problems and fractional optimal control problems.

In this paper, the method of steps is used to convert the VO-DMs (1.1), (1.2) , (1.3) and (1.4) into non delay ones. After, A new shifted fractional Gegenbauer operational matrix (SFGOM) of Caputo VO derivatives in conjunction with the spectral collocation method is used to handle the resulting equations. As a result, systems of algebraic equations are obtained, these systems are easily solved using an iterative process.

This paper is arranged as follows. In Section 2, we present some preliminaries of VO calculus and SFGPs. In Section 3, the SFGOM of the VO derivative is derived and the procedure solution of the present technique is introduced. In Section 4, some illustrative problems are offered. Section 5 ends the paper with a conclusion.

Preliminaries and Definitions:

Caputo variable order derivative

For physical modeling Combira [13] derived the following VO Caputo differential operator

$$D_t^{\nu(t)} y(t) = \frac{1}{\Gamma\left(1-\nu(t)\right)} \int_{0^+}^{t} (t-\xi)^{-\nu(t)} y'(\xi) d\xi + \frac{y(0^+) - y(0^-)}{\Gamma\left(1-\nu(t)\right)} t^{-\nu(t)}, \qquad (2.1)$$

where $y(t) \in C^1[0,L]$ is a continuous differentiable function, $\nu(t)$ be a positive continuous bounded and $0 < \nu(t) \le 1, y'$ is the first integer derivative $y(0^+)$, and $y(0^-)$ are the right-hand and left-hand limits of $y(t)$ at $t = 0,$ respectively. If the beginning time is considered in the ideal situation, then definition (2.1) is reduced to

$$D_t^{\nu(t)} y(t) = \frac{1}{\Gamma\left(1-\nu(t)\right)} \int_{0^+}^{t} (t-\xi)^{-\nu(t)} y'(\xi) d\xi. \qquad (2.2)$$

From definition (2.1), the following formula is established

$$D_t^{\nu(t)} t^k = \begin{cases} \dfrac{\Gamma(k+1)}{\Gamma(k+1-\nu(t)} t^{k-\nu(t)}, & for\ k \in N_0\ and\ k \ge m, \\ 0, & for\ k \in N_0\ and\ k < m, \end{cases}$$

where $N_0 = N \cup \{0\}$.

Mathematical preliminaries of Gegenbauer polynomials

Gegenbauer polynomials

The GPs $C_j^{(\alpha)}(x),$ of degree $j \in Z^+,$ and defined with the parameter $\alpha > \frac{-1}{2}$ are a sequence of real polynomials in the finite domain $[-1,1].$ They are a family of orthogonal polynomials which has many applications [17].

- A suitable standardization of the GPs $C_j^{(\alpha)}(x)$ dates back to Doha [16], where the GPs can be represented by

  $$C_j^{(\alpha)}(x) = \frac{j!\,\Gamma\left(\alpha+\frac{1}{2}\right)}{\Gamma\left(j+\alpha+\frac{1}{2}\right)} P_j^{\left(\alpha-\frac{1}{2},\alpha-\frac{1}{2}\right)}(x), j = 0,1,2,\ldots\ldots,$$

  or equivalently

  $$C_j^{(\alpha)}(1) = 1, \quad j = 0,1,2,\ldots\ldots,$$

  where $P_j^{\left(\alpha-\frac{1}{2},\alpha-\frac{1}{2}\right)}(x)$ are the Jacobi polynomials (JPs) of degree j.

- The GPs' analytical version is provided by,

$$C_j^{(\alpha)}(x) = \sum_{k=0}^{j} (-1)^{j-k} \frac{j!\,\Gamma\left(\alpha+\frac{1}{2}\right)\Gamma(j+k+2\alpha)}{\Gamma\left(k+\alpha+\frac{1}{2}\right)\Gamma(j+2\alpha)(j-k)!\,k!}\left(\frac{x+1}{2}\right)^k \tag{2.4}$$

- The following handy recurrence equation can be used to create the GPs

$$(j+2\alpha)C_{j+1}^{(\alpha)}(x) = 2(j+\alpha)xC_j^{(\alpha)}(x) - jC_{j-1}^{(\alpha)}(x), \qquad j \geq 1. \tag{2.5}$$

- The orthogonality relation is satisfied by the GPs

$$\langle C_i^{(\alpha)}(x), C_j^{(\alpha)}(x)\rangle = \int_{-1}^{1} C_i^{(\alpha)}(x) C_j^{(\alpha)}(x)\,\omega^{(\alpha)}(x)dx = \lambda_j^{(\alpha)}\delta_{i,j},$$

  $\omega^{(\alpha)}(x)$ denotes the weight function, it is an even function that is determined by the relation

  $\omega^{(\alpha)}(x) = (1-x^2)^{\alpha-\frac{1}{2}}$,

  and

$$\lambda_j^{(\alpha)} = \left\| C_j^{(\alpha)}(x)\right\|^2 = \frac{2^{2\alpha-1} j!\,\Gamma^2(\alpha+\frac{1}{2})}{(j+\alpha)\Gamma(j+2\alpha)}, \tag{2.6}$$

  is the normalization factor and $\delta_{i,j}$ represents the Kronecker delta function.

- Now, we construct the fractional Gegenbauer–Gauss quadrature rule, for any $f(x) \in L^2_{\omega^{(\alpha)}(x)}[-1,1]$,

$$\int_{-1}^{1} f(x)\,\omega^{(\alpha)}(x)dx \approx \sum_{j=0}^{N} \varpi_{N,j}^{(\alpha)} f(x_j),$$

  where $\varpi_{N,j}^{(\alpha)}, j = 0,1,\dots,N,$ are the corresponding Christoffel numbers defined by

$$(\varpi_{N,j}^{(\alpha)})^{-1} = \sum_{l=0}^{N} \left(\lambda_l^{(\alpha)}\right)^{-1} (C_l^{(\alpha)}(x_j))^2, \forall j \tag{2.7}$$

  where $\lambda_l^{(\alpha)}$ getting from the relation (2.6).

Shifted fractional Gegenbauer polynomials

- The explicit analytic form of the SFGPs is given as

$$C_{t_0,t_f,j}^{(\alpha,\mu)}(t) = \sum_{k=0}^{j} (-1)^{j-k} \frac{j!\,\Gamma\left(\alpha+\frac{1}{2}\right)\Gamma(j+k+2\alpha)}{\Gamma\left(k+\alpha+\frac{1}{2}\right)\Gamma(j+2\alpha)(j-k)!\,k!}\left(\frac{t-t_0}{t_f-t_0}\right)^{\mu k} \tag{2.8}$$

- For $\mu = 1$, the SFGPS reduced to the shifted GPs in the interval $[t_0, t_f]$.
- The orthogonal relation of SFGPs is

$$\int_{t_0}^{t_f} C_{t_0,t_f,i}^{(\alpha,\mu)}(t) C_{t_0,t_f,j}^{(\alpha,\mu)}(t)\,\omega_{t_0,t_f}^{(\alpha,\mu)}(t)dt = \lambda_{t_0,t_f,j}^{(\alpha,\mu)}\delta_{i,j}, \tag{2.9}$$

  where $\omega_{t_0,t_f}^{(\alpha,\mu)}(t) = (t^\mu - t_0)^{\alpha-\frac{1}{2}}(t_f - t^\mu)^{\alpha-\frac{1}{2}}$ is the weight function, and

$$\lambda_{t_0,t_f,j}^{(\alpha,\mu)} = \left(\frac{(t_f-t_0)^\mu}{2}\right)^{2\alpha} \frac{2^{2\alpha-1} j!\,\Gamma^2(\alpha+\frac{1}{2})}{(j+\alpha)\Gamma(j+2\alpha)},$$

  is the normalization factor and $\delta_{i,j}$ represents the Kronecker delta function.

- The SFGPs comprise unlimited number of orthogonal polynomials, among them the shifted fractional order CPs of the first kind $T_{t_0,t_f,i}^{(\mu)}(t) \equiv C_{t_0,t_f,i}^{(0,\mu)}(t)$, the shifted fractional order CPs of the second kind $U^{(\mu)}_{t_0,t_f,i}(t) \equiv (i+1)C^{(1,\mu)}_{t_0,t_f,i}(t)$, and the shifted fractional order LPs $L^{(\mu)}_{t_0,t_f,i}(t) \equiv C^{(\frac{1}{2},\mu)}_{t_0,t_f,i}(t)$.

- We denote the zeros of the SFGPs $C^{(\alpha,\mu)}_{t_0,t_f,N+1}(t)$ by $t^{(\mu)}_{t_0,t_f,N,j}, j = 0, \ldots, N$.

$$t^{(\mu)}_{t_0,t_f,N,j} = \left(\frac{t_{N,j}+1}{2}\right)^{\frac{1}{\mu}},$$

where $t_{N,j}, j = 0, \ldots, N$ are the zeros of $C^{(\alpha)}_{N+1}(t)$. By using the regular Gegenbauer–Gauss quadrature, the fractional Gegenbauer-Gauss quadrature rule can be established, for any $f\left(\left(\frac{t+1}{2}\right)^{\frac{1}{\mu}}\right) \in P_{2N+1}(t_0, t_f)$, we have

$$\int_{t_0}^{t_f} \omega^{(\alpha,\mu)}_{t_0,t_f}(t) f(t)\,dt = \left(\frac{(t_f - t_0)^{\mu}}{2}\right)^{2\alpha} \int_{-1}^{1} \omega^{(\alpha)}(t) f\left(\left(\frac{t+1}{2}\right)^{\frac{1}{\mu}}\right) dt,$$

$$= \left(\frac{(t_f - t_0)^{\mu}}{2}\right)^{2\alpha} \sum_{j=0}^{N} \varpi^{(\alpha)}_{N,j}\, f\left(\left(\frac{t_{N,j}+1}{2}\right)^{\frac{1}{\mu}}\right),$$

$$= \sum_{j=0}^{N} \varpi^{(\alpha,\mu)}_{t_0,t_f N,j}\, f\left(t^{(\mu)}_{t_0,t_f,N,j}\right),$$

where $\varpi^{(\alpha,\mu)}_{t_0,t_f N,j}$ are the Christoffel numbers of the SFGPs that are given by

$$\varpi^{(\alpha,\mu)}_{t_0,t_f N,j} = \left(\frac{(t_f - t_0)^{\mu}}{2}\right)^{2\alpha} \varpi^{(\alpha)}_{N,j}, j = 0, \ldots, N, \quad (2.10)$$

where $\varpi^{(\alpha)}_{N,j}$ are defined by (2.7).

- The square integrable function $y(t) \in [t_0, t_f]$ can be approximated by SFGPs as

$$y(t) = \sum_{j=0}^{\infty} \tilde{y}_{t_0,t_f,j}\, C^{(\alpha,\mu)}_{t_0,t_f,j}(t),$$

where the coefficients $\tilde{y}_{t_0,t_f,j}$ are determined by

$$\tilde{y}_{t_0,t_f,j} = (\lambda^{(\alpha,\mu)}_{t_0,t_f,j})^{-1} \int_{t_0}^{t_f} y(t)\ \omega^{(\alpha,\mu)}_{t_0,t_f}(t) C^{(\alpha,\mu)}_{t_0,t_f,j}(t)\,dt, j = 0, 1, \ldots. \quad (2.11)$$

If we approximat $y(t)$e by the first (N+1)-terms, then we can write

$$y(t) \simeq y_N(t) = \sum_{j=0}^{N} \tilde{y}_{t_0,t_f,j}\, C^{(\alpha,\mu)}_{t_0,t_f,j}(t). \quad (2.12)$$

The approximation of function $y(t)$ can be written in the vector form as

$$y(t) \simeq Y^T \Phi(t), \quad (2.13)$$

Where $Y^T = [Y_0, Y_1, \ldots, Y_N]$ is the shifted Gegenbauer coefficient vector, and

$$\Phi(t) = [C^{(\alpha,\mu)}_{t_0,t_f,0}(t), C^{(\alpha,\mu)}_{t_0,t_f,1}(t), \ldots, C^{(\alpha,\mu)}_{t_0,t_f,N}(t)]^T = AT_N(t), \quad (2.14)$$

is the SFG vector.

Error estimation

Let $\mathbb{C}^{(\alpha,\mu)}_{t_0,t_f,N} = Span\{C^{(\alpha,\mu)}_{t_0,t_f,0}(t), C^{(\alpha,\mu)}_{t_0,t_f,1}(t), \dots, C^{(\alpha,\mu)}_{t_0,t_f,N}(t)\}$ and $y(t)$ be an arbitrary element in $L^2_{\omega^{(\alpha,\mu)}_{t_0,t_f}}[t_0,t_f]$. Sinc $\mathbb{C}^{(\alpha,\mu)}_{t_0,t_f,N}$e is a finite dimensional vector space, $y(t)$ has the unique best approximation out of $\mathbb{C}^{(\alpha,\mu)}_{t_0,t_f,N}$ like $y_N(t) \in \mathbb{C}^{(\alpha,\mu)}_{t_0,t_f,N}$, such that

$$\forall\, g(t) \in \mathbb{C}^{(\alpha,\mu)}_{t_0,t_f,N}, \qquad \|y(t) - y_N(t)\|_2 \le \|y(t) - g(t)\|_2,$$

where $\|y(t)\| = \sqrt{\langle y(t), y(t)\rangle}$.

Theorem 2.1

Suppose that the function $D^{k\mu}y(t) \in C[t_0,t_f], \mu \in [0,1]$ for $k = 0,1,\dots,N-1$. $y_N(t)$ is the best approximation to $y(t)$ from $\mathbb{C}^{(\alpha,\mu)}_{t_0,t_f,N}$ then the error bound is presented as follow

$$\|y(t) - y_N(t)\|_2 \le \frac{H_\mu}{\Gamma((N+1)\mu+1)} \sqrt{\sum_{j=0}^{N} \varpi^{(\alpha,\mu)}_{t_0,t_f,N,j} \left(t^{(\mu)}_{t_0,t_f,N,j} - t_0\right)^{2(N+1)\mu}}, \tag{2.15}$$

where $\|D^{(N+1)\mu}y(t)\| \le H_\mu, t \in [t_0,t_f]$.

Proof

We consider the generalized Taylor formula [26]

$$y(t) = \sum_{i=0}^{N} \frac{t^{i\mu}}{\Gamma(i\mu+1)} D^{i\mu}y(t_0) + \frac{D^{(N+1)\mu}y(\xi)}{\Gamma((N+1)\mu+1)}(t-t_0)^{(N+1)\mu},$$

with $\xi \in [t_0,t], \forall t \in [t_0,t_f]$. Let

$$P_N(t) = \sum_{i=0}^{N} \frac{t^{i\mu}}{\Gamma(i\mu+1)} D^{i\mu}y(t_0).$$

Then,

$$|y(t) - P_N(t)| = \left|\frac{D^{(N+1)\mu}y(\xi)}{\Gamma((N+1)\mu+1)}(t-t_0)^{(N+1)\mu}\right|.$$

Because of $y_N(t)$ is the best square approximation function of $y(t)$, we can gain

$$\|y(t) - y_N(t)\|_2^2 \le \|y(t) - P_N(t)\|_2^2,$$

$$= \int_{t_0}^{t_f} \omega^{(\alpha,\mu)}_{t_0,t_f}(t)\big(y(t) - P_N(t)\big)^2 dt,$$

$$= \int_{t_0}^{t_f} \omega^{(\alpha,\mu)}_{t_0,t_f}(t)\left(\frac{D^{(N+1)\mu}y(\xi)}{\Gamma((N+1)\mu+1)}(t-t_0)^{(N+1)\mu}\right)^2 dt,$$

$$= \frac{(H_\mu)^2}{\Gamma^2((N+1)\mu+1)} \int_{t_0}^{t_f} (t^\mu - t_0)^{\alpha-\frac{1}{2}}(t_f - t^\mu)^{\alpha-\frac{1}{2}}(t-t_0)^{2(N+1)\mu} dt,$$

$$= \frac{(H_\mu)^2}{\Gamma^2((N+1)\mu+1)} \sum_{j=0}^{N} \varpi_{t_0 t_f N,j}^{(\alpha,\mu)} \left(t_{t_0 t_f,N,j}^{(\mu)} - t_0\right)^{2(N+1)\mu},$$

where $\varpi_{t_0 t_f N,j}^{(\alpha,\mu)}$ are the Christoffel numbers of the SFGPs which are given by (2.10). By applying the square roots, the proof is completed.

The Methodology

Shifted Fractional Gegenbauer Operational Matrix (SFGOM)

In this subsection, we will deduced the OM of the VO fractional derivative for the SFG vector $\Phi(t)$, where $t \in [t_0, t_f]$.

From relation (2.8) by using VO Caputo fractional derivative, we can write

$$D_t^{\nu(t)} C_{t_0 t_f,i}^{(\alpha,\mu)}(t) = \sum_{k=0}^{i} (-1)^{i-k} \frac{i!\,\Gamma\left(\alpha+\frac{1}{2}\right)\Gamma(i+k+2\alpha)}{\Gamma\left(k+\alpha+\frac{1}{2}\right)\Gamma(i+2\alpha)(i-k)!\,k!\,(t_f-t_0)^{\mu k}} D_t^{\nu(t)} (t-t_0)^{\mu k},$$

$$= \sum_{k=\lceil \nu(t) \rceil}^{i} (-1)^{i-k} \frac{i!\,\Gamma\left(\alpha+\frac{1}{2}\right)\Gamma(i+k+2\alpha)\Gamma(\mu k+1)}{\Gamma\left(k+\alpha+\frac{1}{2}\right)\Gamma(i+2\alpha)(i-k)!\,k!\,\left(t_f-t_0\right)^{\mu k}\Gamma(\mu k+1-\nu(t))} (t-t_0)^{\mu k-\nu(t)}, \quad (3.1)$$

The function $(t-t_0)^{\mu k-\nu(t)}$ can be written as a series of N + 1 terms of SFG polynomials,

$$(t-t_0)^{\mu k-\nu(t)} = \sum_{j=0}^{N} \tilde{t}_j C_{t_0 t_f,j}^{(\alpha,\mu)}(t), \quad (3.2)$$

where

$$\tilde{t}_j = \frac{1}{\lambda_{t_0 t_f,j}^{(\alpha,\mu)}} \int_{t_0}^{t_f} (t^\mu - t_0)^{\alpha-\frac{1}{2}} \left(t_f - t^\mu\right)^{\alpha-\frac{1}{2}} (t-t_0)^{\mu k-\nu(t)} C_{t_0 t_f,j}^{(\alpha,\mu)}(t)\,dt, \quad (3.3)$$

$$\lambda_{t_0 t_f,j}^{(\alpha,\mu)} = \frac{\left(t_f-t_0\right)^{2\alpha\mu} j!\,\Gamma^2(\alpha+\frac{1}{2})}{2(j+\alpha)\Gamma(j+2\alpha)}, \quad (3.4)$$

$$C_{t_0 t_f,j}^{(\alpha,\mu)}(t) = \sum_{f=0}^{j} (-1)^{j-f} \frac{j!\,\Gamma\left(\alpha+\frac{1}{2}\right)\Gamma(j+f+2\alpha)}{\Gamma\left(f+\alpha+\frac{1}{2}\right)\Gamma(j+2\alpha)(j-f)!\,f!} \left(\frac{t-t_0}{t_f-t_0}\right)^{\mu f}, \quad (3.5)$$

$$\tilde{t}_j = \sum_{f=0}^{j} \frac{2(-1)^{j-f}(j+\alpha)\Gamma(j+f+2\alpha)}{\Gamma\left(f+\alpha+\frac{1}{2}\right)\Gamma\left(\alpha+\frac{1}{2}\right)(j-f)!\,f!\,\left(t_f-t_0\right)^{\mu f+2\alpha\mu}} \int_{t_0}^{t_f} (t-t_0)^{\mu k-\nu(t)+\mu f} (t^\mu - t_0)^{\alpha-\frac{1}{2}} \left(t_f - t^\mu\right)^{\alpha-\frac{1}{2}} dt,$$

By using (3.4) and (3.5) into (3.3), we have

$$\tilde{t}_j = \sum_{f=0}^{j} \frac{2(-1)^{j-f}(j+\alpha)\Gamma(j+f+2\alpha)}{\Gamma\left(f+\alpha+\frac{1}{2}\right)\Gamma\left(\alpha+\frac{1}{2}\right)(j-f)!\,f!\,\left(t_f-t_0\right)^{\mu f+2\alpha\mu}} \left(\frac{t_f-t_0}{2}\right)^{2\alpha} \sum_{y=0}^{N} \varpi_{t_0 t_f,y}^{(\alpha,\mu)} (t_{t_0 t_f,y} - t_0)^{\mu k-\nu(t)+\mu f},$$

$$\tilde{t}_j = \sum_{f=0}^{j} \frac{2^{1-2\alpha}(-1)^{j-f}(j+\alpha)\Gamma(j+f+2\alpha)}{\Gamma\left(f+\alpha+\frac{1}{2}\right)\Gamma\left(\alpha+\frac{1}{2}\right)(j-f)!\,f!\,\left(t_f-t_0\right)^{\mu f+2\alpha\mu-2\alpha}} \sum_{y=0}^{N} \varpi_{t_0 t_f,y}^{(\alpha,\mu)} (t_{t_0 t_f,y} - t_0)^{\mu k-\nu(t)+\mu f}, \quad (3.6)$$

Now, by employing (3.1), (3.2), (3.3), (3.4), (3.5) and (3.6) we obtain:

$$D_t^{\nu(t)} C_{t_0,t_f,i}^{(\alpha,\mu)}(t)$$
$$= \sum_{k=\lceil \nu(t)\rceil}^{i} \sum_{j=0}^{N} (-1)^{i-k} \frac{i!\,\Gamma\left(\alpha+\frac{1}{2}\right)\Gamma(i+k+2\alpha)\Gamma(\mu k+1)}{\Gamma\left(k+\alpha+\frac{1}{2}\right)\Gamma(i+2\alpha)(i-k)!\,k!\,(t_f-t_0)^{\mu k}\Gamma(\mu k+1-\nu(t))} \tilde{t}_j\, C_{t_0,t_f,j}^{(\alpha,\mu)}(t),$$
$$= \sum_{j=0}^{N}\left(\sum_{k=\lceil \nu(t)\rceil}^{i} \xi_{i,j,k}\right) C_{t_0,t_f,j}^{(\alpha,\mu)}(t), \quad i=0,1,\dots,N \tag{3.7}$$

where $\xi_{i,j,k}$ is given by

$$\xi_{i,j,k} = \Xi \times \Upsilon, \tag{3.8}$$

where

$$\Xi = \sum_{k=\lceil \nu(t)\rceil}^{i} (-1)^{i-k} \frac{i!\,\Gamma\left(\alpha+\frac{1}{2}\right)\Gamma(i+k+2\alpha)\Gamma(\mu k+1)}{\Gamma\left(k+\alpha+\frac{1}{2}\right)\Gamma(i+2\alpha)(i-k)!\,k!\,(t_f-t_0)^{\mu k}\Gamma(\mu k+1-\nu(t))},$$

$$\Upsilon = \sum_{f=0}^{j} \frac{2^{1-2\alpha}(-1)^{j-f}(j+\alpha)\Gamma(j+f+2\alpha)}{\Gamma\left(f+\alpha+\frac{1}{2}\right)\Gamma\left(\alpha+\frac{1}{2}\right)(j-f)!\,f!\,(t_f-t_0)^{\mu f+2\alpha\mu-2\alpha}} \sum_{y=0}^{N} \varpi_{t_0,t_f,y}^{(\alpha,\mu)} (t_{t_0,t_f,y}-t_0)^{\mu k-\nu(t)+\mu f}.$$

where $\varpi_{t_0,t_f,y}^{(\alpha,\mu)}$ is the Christoffel numbers which is determined by (2.10).

So, in a vector form, the VO fractional derivative SFGP is written as

$$D_t^{\nu(t)}\Phi(t) \simeq P^{\nu(t)}\Phi(t), \tag{3.9}$$

where $t \in [t_0, t_f]$ and $P^{\nu(t)}$ is called the OM of fractional derivative of order $\nu(t)$ in the Caputo sense.

Procedure solution of the proposed Technique

VO-DDEs with initial function

To solve the VO-DDEs, a numerical strategy based on the method of steps in addition to the SFGOMs combined with the spectral collocation approach will be used. Firstly, for $t \in [0,\tau]$ the VO-DDEs (1.1) and (1.2) are converted to VO non-delay differential equation by applying the method of steps, as

$$D_t^{\nu(t)} y(t) = F\big(t, y(t), g(t-\tau)\big), 0 \le t \le t_f, \tag{3.10}$$

with the IC,

$$y(0) = g(0), \tag{3.11}$$

Now in order to solve (3.10) and (3.11) by using the OM of SFGPs, we approximate $y(t)$, $D_t^{\nu(t)} y(t)$ in terms of SFGPs as follows

$$y(t) \simeq y_N(t) = \sum_{j=0}^{N} \tilde{y}_{t_0,t_f,j}\, C_{t_0,t_f,j}^{(\alpha,\mu)}(t) = Y^T\Phi(t), \tag{3.12}$$

$$D_t^{\nu(t)} y_N(t) \simeq D_t^{\nu(t)}\left(\sum_{j=0}^{N} \tilde{y}_{t_0,t_f,j}\, C_{t_0,t_f,j}^{(\alpha,\mu)}(t)\right) = Y^T D_t^{\nu(t)}(\Phi(t)), \tag{3.13}$$

By using the VO-OM (3.9),

$$D_t^{\nu(t)} y_N(t) = Y^T P^{\nu(t)}\, \Phi(t). \tag{3.14}$$

By substituting from (3.12), (3.13) and (3.14) into (3.10) and (3.11) we get

$$Y^T P^{\nu(t)}\, \Phi(t) = \mathrm{F}\big(\mathrm{t}, Y^T \Phi(t), \mathrm{g}(\mathrm{t}-\tau)\big), \tag{3.15}$$

with the IC,

$$Y^T \Phi(0) = g(0). \tag{3.16}$$

By collocating (3.15) and (3.16) at the t_k nodes, the following collocating scheme is obtained

$$Y^T P^{\nu(t_k)}\, \Phi(t_k) = \mathrm{F}\big(t_k, Y^T \Phi(t_k), \mathrm{g}(t_k-\tau)\big), \tag{3.17}$$

with the IC,

$$Y^T \Phi(0) = g(0). \tag{3.18}$$

So system of (N+1) algebraic equations with the necessary SFG coefficients $\tilde{y}_{t_0 t_f, j}, j = 0,1, \dots, N$ is attained, which can be solved using an appropriate iterative process. As a result, the approximate solution $y_N(t)$ can be determined over the interval $[0, \tau]$.

After, by assuming that $g(t) = y_N(t)$ and applying the same strategy the solution over the interval $[\tau, 2\tau]$ is obtained, and so on by repeating the process, the solutions over the other intervals $[2\tau, 3\tau], [3\tau, 4\tau],$ until we pass the time t_f are computed.

VO-VDIDES with initial function

In order to solve the VO-VDIDES for $t \in [0, \tau]$ the VO-VDIDES (1.3) and (1.4) are converted to VO Volterra non-delay integro-differential equation by applying the method of steps, as

$$D_t^{\nu(t)} y(t) = f_1\big(t, y(t)\big) + \int_{t_0}^{t} K\big(q(\xi-\tau)\big) d\xi, \quad t_0 \le t \le t_f, \;\; 0 < \nu(t) \le 1, \tag{3.19}$$

with the IC,

$$y(t_0) \;= y_0, \tag{3.20}$$

Now in order to solve (3.19) and (3.20) by using the OM of SFGPs, we approximate $y(t),\, D_t^{\nu(t)} y(t)$ in terms of SFGPs as follows

$$y_N(t) \simeq \sum_{j=0}^{N} \tilde{y}_{t_0 t_f, j}\, C_{t_0 t_f, j}^{(\alpha,\mu)}(t) = Y^T \Phi(t), \tag{3.21}$$

$$D_t^{\nu(t)} y_N(t) \simeq D_t^{\nu(t)} \left(\sum_{j=0}^{N} \tilde{y}_{t_0 t_f, j}\, C_{t_0 t_f, j}^{(\alpha,\mu)}(t) \right) = Y^T\, D_t^{\nu(t)}\big(\Phi(t)\big), \tag{3.22}$$

By using the VO-OM (3.9), Eq. (3.22) can be written as

$$D_t^{\nu(t)} y_N(t) = Y^T P^{\nu(t)}\, \Phi(t). \tag{3.23}$$

Substitute from (3.21), (3.2) and (3.23) into (3.19) and (3.20), gives

$$Y^T P^{\nu(t)}\, \Phi(t) = f_1\big(t, Y^T \Phi(t)\big) + \int_{t_0}^{t_f} K\big(q(\xi-\tau)\big) d\xi, \tag{3.24}$$

with the IC

$$Y^T \Phi(0) = y_0. \tag{3.25}$$

This results system of (N+1) algebraic equations with the necessary SFG coefficients $\tilde{y}_{t_0,t_f,j}, j = 0,1,\dots,N$ which can be solved using any iterative process. As a result, the estimated solution $y_N(t)$ can be computed over the interval $[0,\tau]$. After, by assuming that $q(t) = y_N(t)$ and applying the same scheme the solution over the interval $[\tau, 2\tau]$ is obtained, and so on by using the same procedure, the solutions over the other intervals $[2\tau, 3\tau], [3\tau, 4\tau],$ until we pass the time t_f are computed.

Results and discussion

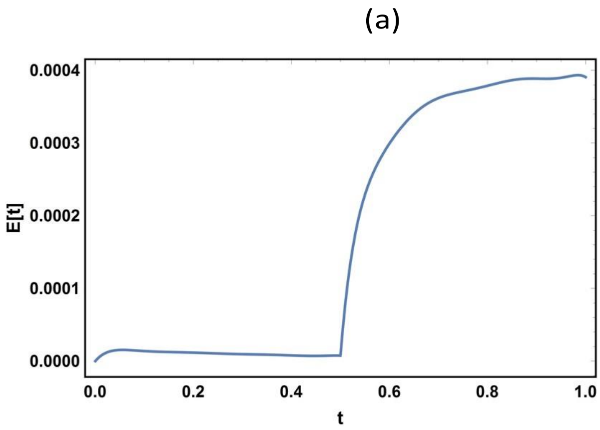
We'll go over five problems in this section to check the applicability and efficiency of the proposed approach. The software was created in Mathematica version 12 using a Dell laptop with an Intel(R) Core(TM) i3 CPU M 370@ 2.40 GHz and 3.00GB RAM configuration. The outcomes of our method are compared to those of other published numerical methodologies in the literature. The results evaluated in this article are calculated via

- The absolute errors (AEs) computed by the relation
$$E(t_i) = |y(t_i) - y_N(t_i)|, \quad 1 \le i \le N,$$
Where $y(t_i)$ and $y_N(t_i)$ represent the ES and the approximate solution (AS), respectively.
- The maximum absolute errors (MAEs) computed by the relation
$$L_\infty = \max_{1\le i\le N}\{E(t_i): \forall\, t_i \in [0, t_f]\}.$$
- Convergence order (CO)
$$CO = \frac{\log\left(\frac{error(N_1)}{error(N_2)}\right)}{\log\left(\frac{N_2}{N_1}\right)},$$
where error(N) is the error associated with a polynomial of degree N.
- The computational time (CPU time) with seconds.

VO-DDEs problems

Problem (1):

Consider the following VO-DDEs,
$$D_t^{\nu(t)} y(t) = -y(t-\tau) - y(t) + F(t), \ \ 0 < \nu(t) \le 1, \quad 0 \le t \le 1,$$
$$y(t) = 0, \quad -\tau \le t \le 0$$
where $\nu(t) = 1 - \frac{t}{2}$,
$$F(t) = \begin{cases} \frac{\Gamma(11)}{\Gamma(11-\nu(t))} t^{10-\nu(t)} + 10^{10}, & 0 \le t \le \tau, \\ \frac{\Gamma(11)}{\Gamma(11-\nu(t))} t^{10-\nu(t)} + 10^{10} + (t-\tau)^{10}, & \tau < t \le 1 \end{cases}$$
and the ES is
$$y(t) = \begin{cases} 0, & \tau < t \le 0, \\ t^{10}, & 0 < t \le 1. \end{cases}$$



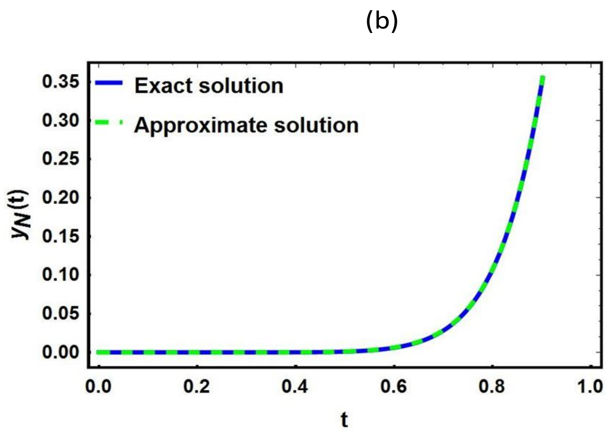


Fig. (1): The numerical results of the proposed technique for problem (1) at N=7, and $\alpha = 0.5, \tau = 0.5, v(t) = 1 - \frac{t}{2}$. (a) The AEs graph of y(t). (b) The graph of ES and AS in [0,1]

Table (1): The AEs of problem (1) at $\mu = 1, \alpha = 0.5$ and different values of N.

t ∈[0,0.5]				
t	$N = 3$	$N = 5$	$N = 7$	$N = 9$
0.0	1.0842×10^{-19}	5.96311×10^{-19}	2.43945×10^{-19}	8.80914×10^{-20}
0.1	6.43967×10^{-4}	1.91959×10^{-4}	1.38223×10^{-5}	1.21526×10^{-7}
0.2	6.18936×10^{-4}	1.47958×10^{-4}	1.1665×10^{-5}	1.02796×10^{-7}
0.3	4.26738×10^{-4}	1.29684×10^{-4}	9.50311×10^{-6}	8.58482×10^{-8}
0.4	4.8735×10^{-4}	9.92666×10^{-5}	7.85919×10^{-6}	7.14791×10^{-8}
0.5	6.28599×10^{-4}	1.15972×10^{-4}	7.58464×10^{-6}	6.47324×10^{-8}
CPU Time	**0.390003**	**11.9653**	**84.5837**	**478.892**
t∈(0.5,1]				
t	$N = 3$	$N = 5$	$N = 7$	$N = 9$
0.6	9.00422×10^{-2}	3.74286×10^{-3}	2.99503×10^{-4}	2.96231×10^{-4}
0.7	5.54744×10^{-2}	1.98243×10^{-3}	3.61941×10^{-4}	3.54816×10^{-4}
0.8	2.12647×10^{-2}	1.83176×10^{-3}	3.78847×10^{-4}	3.78014×10^{-4}
0.9	4.62955×10^{-2}	1.10873×10^{-3}	3.88693×10^{-4}	3.87305×10^{-4}
1.0	4.68614×10^{-2}	1.73337×10^{-3}	3.90701×10^{-4}	3.89935×10^{-4}
CPU Time	**0.374402**	**10.7329**	**88.1094**	**1159.03**

Table (2): The AEs of problem (1) at $\mu = 1, N = 7$ and different values of α.

t ∈[0,0.5]				
t	$\alpha = -0.1$	$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
0.0	1.49078×10^{-19}	6.77626×10^{-20}	2.43945×10^{-19}	2.81893×10^{-18}
0.1	1.05739×10^{-5}	1.11304×10^{-5}	1.38223×10^{-5}	8.72698×10^{-4}
0.2	9.12765×10^{-6}	9.55903×10^{-6}	1.1665×10^{-5}	5.54573×10^{-4}
0.3	7.32909×10^{-6}	7.70015×10^{-6}	9.50311×10^{-6}	3.51214×10^{-4}
0.4	6.13383×10^{-6}	6.42682×10^{-6}	7.85919×10^{-6}	3.49391×10^{-4}
0.5	5.78045×10^{-6}	6.08383×10^{-6}	7.58464×10^{-6}	4.55005×10^{-4}
t∈(0.5,1]				
t	$\alpha = -0.1$	$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
0.6	2.78292×10^{-4}	2.83073×10^{-4}	2.99503×10^{-4}	2.78292×10^{-4}
0.7	3.5277×10^{-4}	3.54745×10^{-4}	3.61941×10^{-4}	3.5277×10^{-4}
0.8	3.68984×10^{-4}	3.71341×10^{-4}	3.78847×10^{-4}	3.68984×10^{-4}
0.9	3.85244×10^{-4}	3.85908×10^{-4}	3.88693×10^{-4}	3.85244×10^{-4}
1.0	3.85626×10^{-4}	3.86796×10^{-4}	3.90701×10^{-4}	3.85626×10^{-4}

Tables 1 and 2 list the numerical results of problem (1), which are graphically depicted in Figs. 1(a-b). In Table 1, we tabulated CPU time at $\mu = 1, \tau = 0.5, \alpha = 0.5, v(t) = 1 - t/2$ and calculated the AEs at different values of N using the SFGPs. The numerical results illustrate the efficiency of the proposed method.

Table 2 also shows the AEs for different α values. In Fig. 1(a), We plot the AEs between the ES and the AS at $\mu = 1, N = 7$ and $\alpha = 0.5$ to demonstrate the convergence of the suggested approach. The harmony between the ES and AS is depicted in Fig. 1(b).

Problem (2): Houseflies model

Consider the following VO-DDEs,

$$D_t^{v(t)} y(t) = -dy(t) + cy(t-\tau)\big(k - czy(t-\tau)\big), \quad 0 < v(t) \le 1, \qquad t > 0,$$
$$y(t) = 160, t \in [-\tau, 0].$$

For $v(t) = 1$, $k = 0.5107, c = 1.81, z = 0.000226, d = 0.147,$ and $\tau = 3$ the ES [26] is

$$y(t) = \begin{cases} \frac{20175}{23} - \frac{16495}{23} e^{\frac{-147\,t}{1000}}, & 0 \le t \le 3, \\ \frac{49221}{19} e^{\frac{-147t+441}{500}} + \frac{16117t}{60} e^{\frac{-147t+441}{1000}} + \frac{11483}{7} - \frac{122101}{17} e^{\frac{-147t}{1000}}, & 3 \le t \le 6. \end{cases}$$

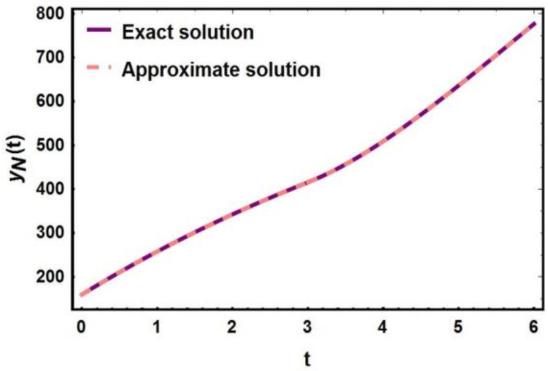


Fig. (2): The numerical results of the proposed technique for problem (2) at N=7, and $\alpha = 0.5,\ \nu(t) = 1.$

Table (3): The AEs of problem (2) at $\mu = 1, \alpha = 1, \nu(t) = \frac{1}{2}$ and distinct values of N.

t∈ [0, 3]			
t	N = 5	N = 7	N = 9
0.0	2.84217×10^{-16}	0	5.68434×10^{-14}
0.6	1.104×10^{-4}	2.8386015×10^{-5}	2.84081×10^{-5}
1.2	1.25377×10^{-4}	5.4347272×10^{-5}	5.43676×10^{-5}
1.8	1.4525×10^{-4}	7.8116295×10^{-5}	7.81377×10^{-5}
2.4	1.59673×10^{-4}	9.9879271×10^{-5}	9.98924×10^{-5}
3.0	1.79408×10^{-4}	1.1980559×10^{-4}	1.19819×10^{-4}
CPU Time	7.61285	75.5981	499.827
t ∈ (3, 6]			
t	N = 5	N = 7	N = 9
3.6	1.65604×10^{-2}	2.9999411×10^{-3}	8.07861×10^{-4}
4.2	1.44811×10^{-2}	2.7245224×10^{-3}	6.67153×10^{-4}
4.8	1.35702×10^{-2}	2.4710056×10^{-3}	5.70991×10^{-4}
5.4	1.21369×10^{-2}	2.2371433×10^{-3}	5.68967×10^{-4}
6.0	1.18317×10^{-2}	2.0194911×10^{-3}	3.85297×10^{-4}
CPU Time	7.78445	89.607	535.817

Table (4): Numerical comparison of the numerical results of problem (2) with Ref. [26] at $\mu = 1,\ \alpha = 1, \nu(t) = 1$ and N = 5

t ∈ [0, 3]							
t	Exact	Present method			Ref. [26]		
	v(t) = 1	v(t) = 1	v(t) = 0.9	v(t) = 0.75	v(t) = 1	v(t) = 0.9	v(t) = 0.75
0.0	160	160	160	160	160	160	160
0.6	220.54544	220.54541	222.677243	226.348405	220.5461	228.2375	241.5848
1.2	275.97949	275.97944	274.683694	272.657319	275.9811	281.6995	289.7538
1.8	326.73368	326.73360	319.732418	309.247175	326.7352	326.0291	322.5902
2.4	373.20309	373.20299	359.663813	340.152297	373.2043	364.9633	350.5524
3.0	415.74944	415.74932	395.371548	366.723465	415.7506	400.3329	376.4743
t ∈ (3, 6]							
t	Exact	Present method			Ref. [26]		
	v(t) = 1	v(t) = 1	v(t) = 0.9	v(t) = 0.75	v(t) = 1	v(t) = 0.9	v(t) = 0.75
3.6	466.46112	466.46193	452.175703	433.517535	476.1586	456.7157	427.6485
4.2	532.81911	532.81977	518.642805	498.134288	543.1325	517.9783	481.7216
4.8	609.31444	609.31501	590.141398	560.283429	616.3306	584.1168	538.8993
5.4	691.66435	691.66492	663.558899	619.670149	695.0837	654.7044	598.8689
6.0	776.57623	776.57661	736.799208	675.989163	778.3961	728.8910	660.7993

Tables 3-4 and Figs 2(a-b) represent the numerical results of problem (2). The AEs of problem (2) at different values of N and $\mu = 1, \alpha = 0.5, \nu(t) = 1$ in addition to CPU time are given in Table 3. Also, Table (4) shows the obtained numerical results of problem (2) at $\alpha = 0.5, \nu(t) = 1$ and N = 5 vs the results in [26], the results illustrate the efficiency and applicability of the proposed method. The AEs graph between the ES and the AS is offered in Fig. 2(a), while Fig. 2(b) depicts the convention between the ES and AS.

Problem (3):

Consider the following VO-DDEs [29],

$$D_t^{\nu(t)} y(t) = y(t-\tau) - y(t) + F(t), \quad 0 < \nu(t) \le 1, \qquad 0 \le t \le 2,$$

$$y(t) = t^2 - t, \quad t \in [-1,0],$$

where $\nu(t) = \frac{\sin(t)}{100}$, $F(t) = \frac{\Gamma(3)}{\Gamma(3-\nu(t))} t^{2-\nu(t)} - \frac{\Gamma(2)}{\Gamma(2-\nu(t))} t^{1-\nu(t)} - (t-\tau)^2 + 2t - \tau - t^2$ and the ES is $y(t) = t^2 - t$.

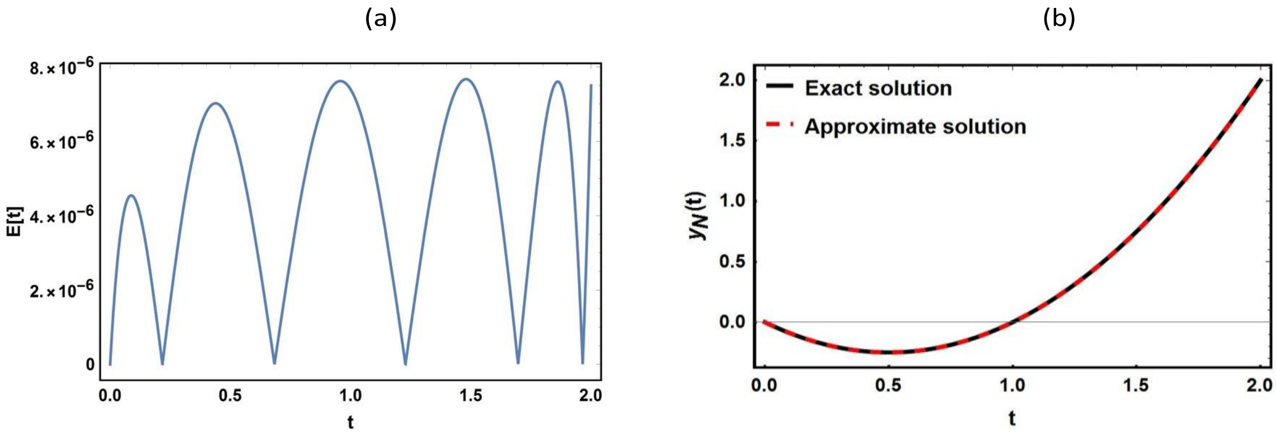


Fig. (3): The numerical behavior of the proposed method on problem (3). **(a)** The AE graph at N=6 and $\alpha = 1$, $v(t) = \sin(t)/100$, $\tau = 0.1e^{-10t}$. **(b)** The graph of the ES and AS at $\tau=0.1e^{-10t}$ on [0,2].

Table (5): MAEs, CPU time and CO of problem (3) at $\mu = 1$, $\alpha = -0.1$, $\tau = 0.01$ and different values of N.

N	MAEs	CPU Time	CO
3	1.76461×10^{-4}	0.358802	---
5	3.64761×10^{-5}	13.1665	3.08607
7	5.25828×10^{-6}	98.6082	5.75635
9	1.64111×10^{-6}	658.527	4.63336

Table (6): The AEs of problem (3) at $\mu = 1$, N = 7, $\tau = 0.01$ and different values of α.

t	$\alpha = -0.1$	$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
0.00	1.88975×10^{-17}	2.02019×10^{-16}	1.70315×10^{-16}	1.44288×10^{-17}
0.25	3.50552×10^{-6}	2.98261×10^{-6}	1.36591×10^{-6}	5.7534×10^{-7}
0.50	8.62483×10^{-8}	2.14619×10^{-7}	4.85245×10^{-7}	4.99196×10^{-7}
0.75	4.94488×10^{-6}	4.22438×10^{-6}	2.13783×10^{-6}	1.21574×10^{-6}
1.00	1.41087×10^{-6}	1.25593×10^{-6}	7.38224×10^{-7}	4.63492×10^{-7}
1.25	4.88779×10^{-6}	4.10385×10^{-6}	1.87043×10^{-6}	9.32063×10^{-7}
1.50	3.08363×10^{-6}	2.89317×10^{-6}	2.02525×10^{-6}	1.38236×10^{-6}
1.75	1.7228×10^{-6}	1.09256×10^{-6}	6.15745×10^{-7}	1.14592×10^{-6}
2.00	3.7694×10^{-6}	4.5457×10^{-6}	7.67698×10^{-6}	9.832×10^{-6}

Table (7): MAEs, CPU time and CO of problem (3) at $\mu = 1, \alpha = 0, \tau = 0.1e^{-10t}$ and different valuesof N.

N	MAEs	CPU Time	CO
3	212789×10^{-4}	2.52722	--
5	3.48173×10^{-5}	46.0671	26.0815
7	4.58242×10^{-6}	441.576	6.02691
9	1.40205×10^{-6}	2184.76	4.71239

Table (8): The AEs of problem (3) at μ = 1, N = 11, $\tau = 0.1e^{-10t}$ and different values of α.

t	$\alpha = -0.1$	$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
0.00	1.01431×10^{-16}	7.27863×10^{-17}	3.66666×10^{-17}	1.71188×10^{-17}
0.25	3.39215×10^{-7}	2.45556×10^{-7}	6.41019×10^{-8}	1.06579×10^{-8}
0.50	4.91078×10^{-7}	3.7008×10^{-7}	1.51761×10^{-7}	5.93153×10^{-8}
0.75	4.00133×10^{-7}	2.89424×10^{-7}	1.25605×10^{-7}	5.46979×10^{-8}
1.00	1.57252×10^{-7}	9.74798×10^{-8}	6.01531×10^{-8}	3.3108×10^{-8}
1.25	1.19054×10^{-7}	1.10341×10^{-7}	8.67002×10^{-9}	1.22976×10^{-8}
1.50	2.59946×10^{-7}	2.08732×10^{-7}	2.86773×10^{-8}	1.85032×10^{-8}
1.75	3.15788×10^{-8}	2.00816×10^{-8}	1.2517×10^{-7}	1.29472×10^{-7}
2.00	3.6878×10^{-7}	5.20462×10^{-7}	1.05377×10^{-6}	1.35213×10^{-6}

The numerical solutions to problem (3) are reported in Tables (5-8) and graphically depicted in Figs. 3(a-b). We tabulated the MAEs, CO, and CPU time at $\mu = 1$, $\tau = 0.01$, $\alpha = -.10$ in Table (5) using the SFGOMs method at various values of N. Also, Table 6 lists the AEs for various values of α. Table (7) introduces the MAEs, CO, and CPU time at $\mu = 1$, $\tau = 0.1e^{-10t}$, $\alpha = 0$ at distinct values of N, μ. In addition, Table (8) presents the AEs at $N = 11$, $\mu = 1$, $\tau = 0.1e^{-10t}$ and various values of α. Fig. 3(a) shows the AEs between the ES and the AS at $\tau = 0.1e^{-10t}$ with $N = 6$, $\mu = 1$ and $\alpha = 1$ to demonstrate the convergence of the suggested approach. The agreement between the ES and AS at $\tau = 0.1e^{-10t}$ is depicted in Fig. 3(b).

VO -VDIDEs problems

Problem (4):

Consider the following VO-VDIDES,

$$D_t^{\nu(t)}y(t) = y(t) + \int_0^t y(\xi - 1)d\xi + \frac{1-e^t}{e} + F(t), \quad 0 < \nu(t) \le 1, \qquad 0 \le t \le 1,$$

$$y(0) = 1$$

$$y(t) = e^t, \qquad t < 0,$$

Where $\nu(t) = \cos(t)\,,\ F(t) = \sum_{k=1}^{\infty} \frac{t^{k-\nu(t)}}{\Gamma(k+1-\nu(t))} - e^t$ and the ES is e^t.

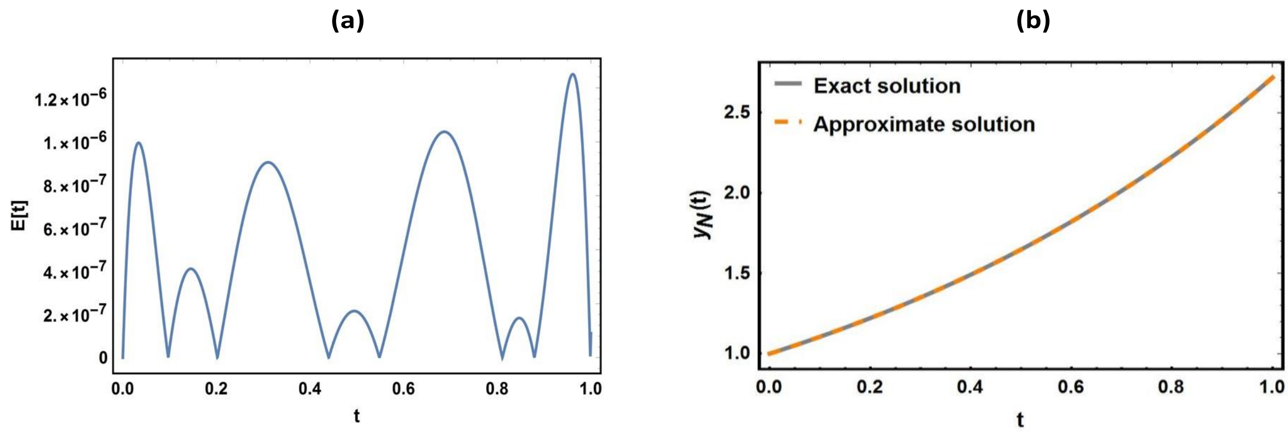


Fig. (4): The numerical behavior of the proposed method on problem (4) at N = 8 and α = 0.5, ν(t)= cos(t), τ=1. **(a)** The AE graph of y(t). **(b)** The graph of the ES and AS on [0,1].

Table (9): MAEs, CPU time and CO of problem (4) at μ = 1, α = 0.5, τ = 1 and different values of N.

N	MAEs	CPU Time	CO
3	1.10307×10^{-3}	1.71601	----
5	1.2496×10^{-5}	20.3893	8.77099
7	2.80965×10^{-6}	114.77	4.43528
9	6.08145×10^{-7}	681.896	6.08959

Table 10: The AEs of problem (4) at μ = 1, N = 8, τ = 1 and distinct values of α.

t	$\alpha = -0.1$	$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
0.0	3.33067×10^{-16}	1.11022×10^{-16}	4.44089×10^{-16}	2.22045×10^{-16}
0.2	8.52819×10^{-7}	7.83197×10^{-7}	2.46773×10^{-8}	9.66403×10^{-7}
0.4	1.03639×10^{-6}	8.29959×10^{-7}	3.49319×10^{-7}	1.55542×10^{-6}
0.6	1.12992×10^{-6}	8.91389×10^{-7}	4.89451×10^{-7}	1.91286×10^{-6}
0.8	1.50214×10^{-6}	1.32508×10^{-6}	1.09032×10^{-7}	1.83164×10^{-6}
1.0	1.39161×10^{-6}	1.21595×10^{-6}	1.13308×10^{-7}	1.03205×10^{-6}

Tables (9-10) and Fig. 4(a-b) show the numerical behavior of problem (4). In Table (9), we tabulate the MAEs, CO and CPU time at μ = 1, α = 0.5 and distinct values of N. The AEs of the problem (4) at distinct values of α, and μ = 1, N = 8 are given in Table 10. Fig. 4(a) depicts the AEs between the ES and the AS, whereas Fig. 4(b) depicts the ES and AS harmony.

Problem (5):

Consider the following VO-VDIDES,

$$D_t^{\nu(t)}y(t) = y(t) + \int_0^t y^2(\xi - 1)d\xi - e^t sinh(t) + F(t), \quad 0 < \nu(t) \leq 1, \qquad 0 \leq t \leq 1,$$

$$y(t) = e^{t+1}, \qquad t < 0,$$

Where $\nu(t) = 1 - \frac{sin(t)}{100}$, $F(t) = e^1 \sum_{k=1}^{\infty} \frac{t^{k-\nu(t)}}{\Gamma(k+1-\nu(t))} - e^{t+1}$ and the ES is e^{t+1}.

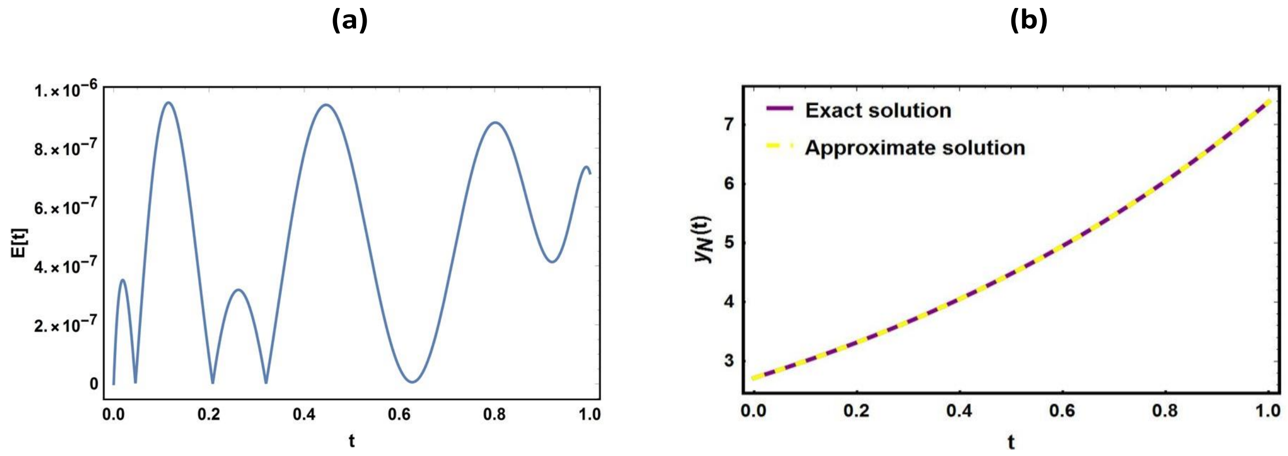


Fig. (5): The numerical behavior of the proposed method for problem (5). **(a)** The AE graph at N = 8 and α = 0, $\nu(t) = 1 - \cos(t)/100$,τ = 1. **(b)** The graph of the ES and AS on [0, 1].

Table (11): MAEs, CPU time and CO of problem (5) at μ = 1,α = 0.5,τ = 1 and different values of N.

N	MAEs	CPU Time	CO
3	2.32496×10^{-3}	1.37281	-----
5	1.3842×10^{-5}	13.6657	10.0303
7	2.38799×10^{-6}	90.6522	5.22259
9	8.21568×10^{-7}	557.844	4.24565

Table (12): The AEs of problem (5) at $\mu = 1$, $N = 9$, $\tau = 1$ and distinct values of α.

t	$\alpha = -0.1$	$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
0.0	4.44089×10^{-16}	8.88178×10^{-16}	1.77636×10^{-15}	4.44089×10^{-16}
0.2	1.52152×10^{-7}	9.37497×10^{-8}	3.36799×10^{-7}	7.82135×10^{-7}
0.4	1.05787×10^{-6}	7.8352×10^{-7}	2.50252×10^{-7}	9.49989×10^{-7}
0.6	2.05106×10^{-7}	5.34808×10^{-8}	7.04283×10^{-7}	1.35686×10^{-6}
0.8	1.21318×10^{-6}	8.86686×10^{-7}	4.64468×10^{-7}	1.45694×10^{-6}
1.0	1.04677×10^{-6}	7.15273×10^{-7}	7.63273×10^{-7}	1.95594×10^{-6}

Tables (11-12) and Figs 5(a-b) show the numerical behavior of problem (5). In Table 11, we tabulate the MAEs, CO and CPU time at $\mu = 1$, $\alpha = 0.5$ and distinct values of N. The AEs of the problem (5) at distinct values of α, and $\mu = 1$, $N = 8$ are given in Table 12. Fig. 5(a) depicts the AEs between the ES and the AS, whereas Fig. 5(b) depicts the ES and AS agreement.

Conclusion

In this paper, a new numerical mechanism has been extracted to find the numerical solutions of the VO- DDEs and VO-VDIDES, this mechanism has the ability to convert the delay problems into non-delay problem by using the method of steps. After, the resulting non-delay equation is approximated by the SFGOM of the VO Caputo derivative in conjunction with the collocation method. By applying this mechanism the aforementioned problems are converted into systems of algebraic equations that have been solved easily using any iteration method. The error analysis of the proposed mechanism was also looked at. Some numerical problems were introduced to illustrate the applicability, accuracy, and efficiency by using a few terms of the SFGPs. From these numerical problems, we have concluded that the SFGPs are easy to compute and use, besides they have rapid convergence. Also, the numerical results demonstrated that the method is effective for every choice of the Gegenbauer parameter α. Besides, the numerical comparisons with other methods in the existing literature clarified the efficiency of the proposed method.

References

[1] **Almeida R., Bastos N.R. and Monteiro M.T.T. (2016):** Modeling some real phenomena by fractional differential equations. *Math. Meth. Appl. Sci.*, 39(16): 4846-4855.

[2] **Almeida R., Bastos N.R., Monteiro M.T.T. (2018):** A fractional Malthusian growth model with variable order using an optimization approach. *Stat. Opt. Inf. Comp.* 6(1): 4–11.

[3] **Abdelkawy M.A., Lopes A.M. and Babatin M.M. (2020):** Shifted fractional Jacobi collocation method for solving fractional functional differential equations of variable order. Chaos, Solitons & Fractals, 134: 109721.

[4] **Ahmed H.F. (2020):** Numerical study on factional differential-algebraic systems by means of Chebyshev Pseudo spectral method. *J. Taibah Uni. Sci.*, 14(1): 1023-1032.

[5] **Abd-Elhameed W.M. (2015):** New Galerkin operational matrix of derivatives for solving Lane-Emden singular-type equations. *The Eur. Phys. J. Plus*, 130(3): 1-12.

[6] **Ahmed H.F. and Melad M.B. (2018):** A New Approach for Solving Fractional Optimal Control Problems Using Shifted Ultraspherical Polynomials. *Pro. Fra. Differ. Appl.*, 4(3): 179-195.

[7] **Ahmed H.F., Moubarak M.R.A. and Hashem W.A. (2021):** Gegenbauer spectral tau algorithm for solving fractional telegraph equation with convergence analysis. *Pramana*, 95(2): 1-16.

[8] **Ahmed H.F. and Melad M.B. (2021):** Efficient method for solving variable-order pantograph models. *Pramana*, 95(4): 1-20.

[9] **Ahmed H.F. and Melad M.B. (2022):** A new numerical strategy for solving nonlinear singular Emden Fowler delay differential models with variable order. *Math. Sci.*

[10] **Bellen A. and Zennaro M. (2013):** Numerical methods for delay differential equations. Oxford university press.

[11] **Bhrawy A.H. and Zaky M.A. (2015):** Numerical simulation for two-dimensional variable-order fractional nonlinear cable equation. *Nonlinear Dynamics*, 80(1): 101-116.

[12] **Bhrawy A.H. and Zaky M. (2016):** Numerical algorithm for the variable-order Caputo fractional functional differential equation. Nonlinear Dynamics, 85(3): 1815-1823.

[13] **Cai W., Chen W., Fang J. and Holm S. (2018):** A survey on fractional derivative modeling of power-law frequency-dependent viscous dissipative and scattering attenuation in acoustic wave propagation. *Appl. Mech. Rev*., 70(3).

[14] **Coimbra C.F. (2003):** Mechanics with variable order differential operators. Annalen der Physik, 12(1112): 692-703.

[15] **Dehghan R. (2019):** A numerical solution of variable order fractional functional differential equation based on the shifted Legendre polynomials. *SeMA J.*, 76(2): 217-226.

[16] **Doha E.H. (1991):** The coefficients of differentiated expansions and derivatives of ultraspherical polynomials. Computers & Mathematics with Applications, 21(2-3): 115-122.

[17] **El-Gindy T., Ahmed H. and Melad M. (2018):** Shifted Gegenbauer operational matrix and its applications for solving fractional differential equations. *J. of the Egy. Math. Soc.*, 26(1): 72-90.

[18] **El-Kalaawy A.A., Doha E.H., Ezz-Eldien S.S., Abdelkawy M.A., Hafez M.A., Amin A.Z.M., Baleanu D. and Zaky M.A (2018):** A computationally efficient method for a class of fractional variational and optimal control problems using fractional Gegenbauer functions. Romanian Reports in Physics, 70(2): 90109.

[19] **El-Gindy T.M., Ahmed H.F., Melad M.B. (2022):** Effective numerical technique for solving variable order integro-differential equations. *J. of Appl. Math. Comp*., 68(4):2823-2855.

[20] **Galue, L., Kalla, S.L. and Al-Saqabi, B.N. (2007):** Fractional extensions of the temperature field problems in oil strata. Appl. Math. Comp., 186(1): 35-44.

[21] **Kumar P. and Chaudhary S.K. (2017):** Analysis of fractional order control system with performance and stability. *Int. J. of Eng. Sci*., 9: 408-416.

[22] **Li X. and Wu B. (2015):** A numerical technique for variable fractional functional boundary value problems. *Appl. Math. Let.*, 43: 108-113.

[23] **Li X., Li H. and Wu B. (2017):** A new numerical method for variable order fractional functional differential equations. *Appl. Math. Let*., 68: 80-86.

[24] **Magin R., Ortigueira M.D., Podlubny I. and Trujillo J. (2011)**: On the fractional signals and systems. Signal Processing, 91(3): pp.350-371.

[25] **Muthukumar P. and Ganesh Priya B. (2017):** Numerical solution of fractional delay differential equation by shifted Jacobi polynomials. *Int. J. of Comp. Math*., 94(3): 471-492.

[26] **Obembe A.D., Hossain M.E. and Abu-Khamsin S.A. (2017)**: Variable-order derivative time fractional diffusion model for heterogeneous porous media. *J. Petro. Sci. Eng.*, 152: 391-405.

[27] **Odibat Z.M. and Shawagfeh N.T. (2007):** Generalized Taylor's formula. *Appl. Math. Comp*., 186(1): 286-293.

[28] **Peykrayegan N., Ghovatmand M. and Noori Skandari M.H. (2021):** On the convergence of Jacobi Gauss collocation method for linear fractional delay differential

[29] equations. *Math. Meth. Appl. Sci*., 44(2): 2237-2253.

[30] **Shaiful E.M. and Utoyo M.I. (2018):** A fractional-order model for HIV dynamics in a two-sex population. *Int. J. of Math. Math. Sci.*

[31] **Smith H.L. (2011):** An introduction to delay differential equations with applications to the life sciences (Vol. 57). New York: Springer.

[32] **Yang J., Yao H. and Wu B. (2018):** An efficient numerical method for variable order fractional functional differential equation. *Appl. Math. Let*., 76: 221-226.

[33] **Zhuang P., Liu F., Anh V. and Turner I. (2009):** Numerical methods for the variable-order fractional advection-diffusion equation with a nonlinear source term. *SIAM J. on Num. Anal*., 47(3): 1760-1781.

[34] **Zhang L. and Li S. (2015):** Regularity of weak solutions of the Cauchy problem to a fractional porous medium equation. Boundary Value Problems, 2015(1): 1-6.

آلية عددية جديدة لحل نموذجين من المعادلات التفاضلية التأخيرية ذات الرتبة المتغيرة

هدى فرغل أحمد[١]، أيمن محمد محمود[٢] و مارينا بنيامين ميلاد[٢]

١ قسم الرياضيات-كلية العلوم- جامعة المنيا- مصر

٢ قسم الرياضيات- كلية العلوم- جامعة الوادى الجديد- مصر

يقدم البحث الحالي آلية عددية فعالة لحل نموذجين مختلفين من المعادلات التفاضلية التأخيرية الخطية / غير الخطية ذات الرتبة المتغيرة؛ تمثل النماذج معادلات تفاضلية تأخيرية ذات رتبة متغيرة و معادلات تفاضلية تكاملية تأخيرية لفولتيرا ذات رتبة متغيرة مع شروط دالية. في الآلية المقترحة ، يتم استخدام طريقة الخطوات لتحويل النماذج التأخيرية ذات الرتبة المتغيرة إلى نماذج غير تأخيرية ذات رتبة متغيرة مع شروط أولية. بعد ذلك، يتم استخدام مصفوفة تشغيلية تفاضلية جديدة ذات رتبة متغيرة لكثيرة حدود جيجينباور الكسرية الملحقة بالتزامن مع طريقة التجميع الطيفية لتحويل النماذج قيد الدراسة الى نظام من المعادلات الجبرية التى يسهل حلها. قمنا أيضا بدراسة تقدير الخطأ للالية المقترحة. يتم التحقق من كفاءة ودقة الالية المقترحة من خلال تطبيقها على العديد من المعادلات التفاضلية التأخيرية سواء بتأخير ثابت أو متغير. تتم مقارنة النتائج العددية التي تم الحصول عليها من الالية المقترحة مع النتائج المنشورة في الأدبيات الأخرى لتوضيح دقة وكفاءة الالية المقترحة.