Filomat 37:30 (2023), 10383–10393
https://doi.org/10.2298/FIL2330383A

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

A computational algorithm for solving linear fractional differential equations of variable order

Khursheed J. Ansari[a,*], Rohul Amin[b], Hafsa[b], Atif Nawaz[b], Fazli Hadi[b]

[a] *Department of Mathematics, College of Science, King Khalid University, Abha, 61413, Saudi Arabia*
[b] *Department of Mathematics, University of Peshawar, Khyber Pakhtunkhwa, 25120, Pakistan*

Abstract. An algorithm for solving a class of linear variable-order fractional differential equations (FDEs) numerically is presented in this paper. We utilized a combination of Caputo fractional derivatives with the Haar wavelet collocation method (HWCM) to numerically solve linear variable order FDEs. Examples are provided to demonstrate the precision of the suggested method. Some examples are provided to demonstrate the effectiveness and precision of HWCM. Additionally, maximum absolute error and mean square root error of each test problem are computed for various numbers of collocation points to demonstrate the validity and application of the suggested method. A comparison of exact and approximative solutions is shown in the figure for different numbers of collocation points.

1. Introduction

A 300-year old mathematical discipline, fractional calculus is a more complex form of standard calculus. As old as the traditional analytic that Newton and Leibniz proposed in 1695, fractional calculus is also a mathematical concept. According to Podlubny [1], a fractional differential equation (FDE) is an equation that uses a fractional derivative (FD). FDEs are used to model a wide range of phenomena in the fields of science, engineering, electric circuits, and mechanical systems. These FDEs are used in a variety of fields, including the diffusion process [2], electrodynamics [3], scattering in quantum physics [4], population models [5], and others [6–8]

Many numerical techniques are described in the literature for solving variable order (VO) FDEs numerically. Based on spectral Tau method Ghoreishi and Yazdani [9] found an effective numerical method for FDEs. They convert FDEs in form of matrix-vector multiplication, they used orthogonal polynomial bases to extend the operational method for the Tau method. They described FDs using Caputo definition of FD. With regard to shifted Jacobi polynomials Doha et al. [10] present a numerical approach for Caputo FDE of any degree shifted Jacobi polynomials. They used a shifted Jacobi tau to proposed a direct solution with nonhomogeneous initial conditions for linear multi-order FDEs. In order to solve the linear multi-order FDEs, they introduced quadrature shifted Jacobi tau approximation. Esmaeili et al. [11] used collocation

2020 *Mathematics Subject Classification*. Primary 34K37.
Keywords. Fractional calculus; Caputo derivatives; Variable-order FDEs; Haar wavelet.
Received: 08 February 2022; Accepted: 04 May 2023
Communicated by Marko Petković
Research supported by King Khalid University.
* Corresponding author: Khursheed J. Ansari
Email addresses: ansari.jkhursheed@gmail.com (Khursheed J. Ansari), raminmath@uop.edu.pk (Rohul Amin), hafsanazir70@gmail.com (Hafsa), aatifnawaz111@gmail.com (Atif Nawaz), fazlihadi543@gmail.com (Fazli Hadi)

method formulated on Müntz polynomials to determine numerical solution of FDEs. Even with a modest number of collocation points, one can get good results using this method. Diethelm et al. [12] developed a predicator-corrector technique to solution of FDEs. This scheme is effective for the used of both linear and nonlinear FDEs and it may be extended to multi-term equations (involving more than one differential operator) too. Albadarneh et al. [13] used fractional finite difference method to develop method for the numerical solutions for linear FDEs of $1 < \alpha < 2$ order. Li [14] produced a fractional integration Chebyshev wavelet operational matrix and used it to solve a nonlinear FDE..

Arikoglu and Ozkol[15] used DTM to FDEs. They proves new theorems related to DTM in FDEs. They solved numerical examples for different types of problems that is Baglay-Torvik, Ricatti and fractional oscillation equations, showing the powerfulness of this technique. Momani and Odibat [16] developed the comparison of methods for solution of linear FDEs. They implement the ADM for the solution of linear FDEs. Both the methods are really effective in finding numerical and analytical solution of FDEs. They also developed a classical method such as the FDM for solution of linear FDEs. Faheem et al. used wavelet collocation method for solution of neutral delay DEs [17], collocation method based on Gegenbauer and Bernoulli wavelets for solution of neutral delay DEs [18], Hermite technique for solution of fractional partial DEs [19].

In this article, HWCM is developed to find solution of linear VO FDEs as

$$D^{\psi(t)}y(t) = y(t)b_1(t) + y'(t)b_2(t) + y''(t)b_3(t) + g(t), \tag{1}$$

where $D^{\psi(t)}y(t)$ is variable order FD in Caputo sense, b_1, b_2 b_3 and g are given functions. Initial conditions (ICs) are $y(0) = \lambda_1$ and $y'(0) = \lambda_2$, $D^{\psi(t)}$ is variable order (VO) derivative. Caputo sense definition of the FD of VO as [1]:

$$D_{0,t}^{\psi(t)}y(t) = \frac{1}{\Gamma(-\psi(t)+1)}\int_0^t \frac{y'(t)}{(-s+t)^{\psi(t)}}ds, \quad \text{where} \quad \psi(t) \in (0,1). \tag{2}$$

The rest of the paper is organized as follows. The preliminary material is included in Section 2. The HWCM technique is explained in Section 3 for the solution of VO linear FDEs. Section 4 contains several examples. Section 6 contains the conclusion.

2. Haar functions

Definition 2.1. *The Haar wavelets, which are made up of piecewise constant functions, are the mathematically simplest of all the wavelet families. These wavelets have the advantage of being analytically integrated any number of times [20]. In Haar wavelet two functions, scaling and mother function, play an essential role.*

Scaling function is [21]

$$h_1(x) = \begin{cases} 1 & \text{if } x \in [a,b), \\ 0 & \text{elsewhere}, \end{cases}$$

and mother wavelet is [21]

$$h_2(x) = \begin{cases} 1 & \text{if } x \in [a, \frac{a+b}{2}), \\ -1 & \text{if } x \in [\frac{a+b}{2}, b), \\ 0 & \text{elsewhere}. \end{cases}$$

Integrating Haar functions from 0 to x will give us

$$P_{i,1}(x) = \int_0^x h_i(s)\,ds, \tag{3}$$

and

$$P_{i,1}(x) = \begin{cases} x-\alpha & \text{if } x \in [\alpha,\beta), \\ \gamma - x & \text{if } x \in [\beta,\gamma), \\ 0 & \text{elsewhere.} \end{cases} \tag{4}$$

Definition 2.2. *HWCM discretizes the* $[a, b]$ *interval as known as collocation point (CPs)[22]*

$$t_i = a + (b-a)\frac{i-1/2}{2M} \quad i = 1,2,\ldots,2M. \tag{5}$$

Remark 2.3. *The integral is calculated by following formula*

$$\int_{\beta_1}^{\beta_2} u(x)dx \approx \frac{\beta_2-\beta_1}{N}\sum_{k=1}^{N} u(x_k) = \frac{\beta_2-\beta_1}{N}\sum_{k=1}^{N} u\left(\beta_1 + (\beta_2-\beta_1)\frac{(k-0.5)}{N}\right). \tag{6}$$

The most recent research employing the Haar approach to numerically solve different problems in the literature is shown in [23–25].

3. Numerical method

For linear VO FDEs solution, we develop HWCM in this section. Applying Caputo derivative to Eq. (1), we get

$$\frac{1}{\Gamma(-\psi(t)+n)}\int_0^t \frac{y^{(n)}(s)ds}{(-s+t)^{-n+\psi(t)+1}} = y(t)b_1(t) + y'(t)b_2(t) + y''(t)b_3(t) + g(t), \tag{7}$$

if $0 < \psi(t) < 1$ then $n = 1$ similarly if we take $1 < \psi(t) < 2$ then $n = 2$. Here, we present method for $n = 2$, so Eq. (7) becomes

$$\frac{1}{\Gamma(-\psi(t)+2)}\int_0^t \frac{y''(s)ds}{(-s+t)^{-2+\psi(t)+1}} = y(t)b_1(t) + y'(t)b_2(t) + y''(t)b_3(t) + g(t),$$

simplifying, we get

$$\frac{1}{\Gamma(-\psi(t)+2)}\int_0^t \frac{y''(s)ds}{(-s+t)^{-1+\psi(t)}} = y(t)b_1(t) + y'(t)b_2(t) + y''(t)b_3(t) + g(t). \tag{8}$$

Now let $y''(t) \in L_2[0,1)$, thus

$$y''(t) = \sum_{i=1}^{N} a_i h_i(t), \tag{9}$$

integrating Eq. (9) from 0 to t, we obtain

$$y'(t) - y'(0) = \sum_{i=1}^{N} a_i P_{i,1}(t),$$

using IC, we get

$$y'(t) - \lambda_2 = \sum_{i=1}^{N} a_i P_{i,1}(t),$$

thus

$$y'(t) = \lambda_2 + \sum_{i=1}^{N} a_i P_{i,1}(t). \tag{10}$$

Integrating Eq. (10) from 0 to t, we get

$$y(t) - y(0) = \lambda_2 t + \sum_{i=1}^{N} a_i P_{i,2}(t),$$

using IC, we obtain

$$y(t) - \lambda_1 = \lambda_2 t + \sum_{i=1}^{N} a_i P_{i,2}(t),$$

thus, we get

$$y(t) = \lambda_1 + \lambda_2 t + \sum_{i=1}^{N} a_i P_{i,2}(t). \tag{11}$$

Using these approximations i.e $y(t)$, $y'(t)$ and $y''(t)$ in Eq. (8), we get

$$\frac{1}{\Gamma(-\psi(t)+2)} \int_0^t \frac{\sum_{i=1}^{N} a_i h_i(s) ds}{(-s+t)^{-1+\psi(t)}} = \left(\sum_{i=1}^{N} a_i P_{i,2}(t) + \lambda_2 t + \lambda_1 \right) b_1(t) + \left(\sum_{i=1}^{N} a_i P_{i,1}(t) + \lambda_2 \right) b_2(t) + \left(\sum_{i=1}^{N} a_i h_i(t) \right) b_3(t) + g(t),$$

so

$$\frac{1}{\Gamma(-\psi(t)+2)} \int_0^t \frac{\sum_{i=1}^{N} a_i h_i(s) ds}{(-s+t)^{-1+\psi(t)}} = \sum_{i=1}^{N} a_i P_{i,2}(t) b_1(t) + \lambda_2 t b_1(t) + \lambda_1 b_1(t) + \sum_{i=1}^{N} a_i P_{i,1}(t) b_2(t) + \lambda_2 b_2(t) + \sum_{i=1}^{N} a_i h_i(t) b_3(t) + g(t),$$

also

$$\frac{1}{\Gamma(-\psi(t)+2)} \int_0^t \frac{\sum_{i=1}^{N} a_i h_i(s) ds}{(-s+t)^{-1+\psi(t)}} - \sum_{i=1}^{N} a_i P_{i,2}(t) b_1(t) - \sum_{i=1}^{N} a_i P_{i,1}(t) b_2(t) - \sum_{i=1}^{N} a_i h_i(t) b_3(t) = \lambda_2 t b_1(t) + \lambda_1 b_1(t) + \lambda_2 b_2(t) + g(t). \tag{12}$$

Now taking common $\sum_{i=1}^{N} a_i$ from Eq. (12), we have

$$\sum_{i=1}^{N} a_i \left(\frac{1}{\Gamma(-\psi(t)+2)} \int_0^t \frac{h_i(s) ds}{(-s+t)^{-1+\psi(t)}} - P_{i,2}(t) b_1(t) - P_{i,1}(t) b_2(t) - h_i(t) b_3(t) \right) = \lambda_2 t b_1(t) + \lambda_1 b_1(t) + \lambda_2 b_2(t) + g(t). \tag{13}$$

CPs t_j for $j = 1, 2, ..., N$ are now placed in Eq. (13), yielding the expression

$$\sum_{i=1}^{N} a_i G(i, j) = \lambda_2 t_j b_1(t_j) + \lambda_1 b_1(t_j) + \lambda_2 b_2(t_j) + g(t_j), \tag{14}$$

$$\text{where} \quad G(i, j) = \frac{1}{\Gamma(-\psi(t_j) + 2)} \int_0^{t_j} \frac{h_i(s)ds}{(-s + t_j)^{-1+\psi(t_j)}} - P_{i,2}(t_j)b_1(t_j) - P_{i,1}(t_j)b_2(t_j) - h_i(t_j)b_3(t_j). \tag{15}$$

Now to determine the value of matrix $G(i, j)$ we use the Lepik[20] technique so for $i = 1$, Eq. (15) becomes

$$G(1, j) = \frac{1}{\Gamma(-\psi(t_j) + 2)} \int_0^{t_j} \frac{h_1(s)ds}{(-s + t_j)^{-1+\psi(t_j)}} - P_{1,2}(t_j)b_1(t_j) - P_{1,1}(t_j)b_2(t_j) - h_1(t_j)b_3(t_j),$$

using value of $h_1(t_j)$ and simplifying, we have

$$\begin{aligned} G(1, j) &= \frac{1}{\Gamma(-\psi(t_j) + 2)} \int_0^{t_j} \frac{ds}{(-s + t_j)^{-1+\psi(t_j)}} - P_{1,2}(t_j)b_1(t_j) - P_{1,1}(t_j)b_2(t_j) - b_3(t_j), \\ &= \frac{1}{\Gamma(-\psi(t_j) + 2)} \frac{t_j^{-\psi(t_j)+2}}{(-\psi(t_j) + 2)} - P_{1,2}(t_j)b_1(t_j) - P_{1,1}(t_j)b_2(t_j) - b_3(t_j). \end{aligned}$$

Case-1 Now for $t_j < 0$ since $h_i(t_j)$=$P_{i,2}(t_j)$=$P_{i,1}(t_j)$=0 thus Eq. (15) becomes

$$G(i, j) = 0.$$

Case-2 For $t_j \in [\alpha, \beta)$ then Eq. (15) becomes

$$G(i, j) = \frac{1}{\Gamma(-\psi(t_j) + 2)} \left(\int_0^{\alpha} \frac{h_i(s)ds}{(-s + t_j)^{-1+\psi(t_j)}} + \int_{\alpha}^{t_j} \frac{h_i(s)ds}{(-s + t_j)^{-1+\psi(t_j)}} \right) - P_{i,2}(t_j)b_1(t_j) - P_{i,1}(t_j)b_2(t_j) - b_3(t_j),$$

using value of $h_i(t_j)$ and simplifying, we obtain

$$G(i, j) = \frac{1}{\Gamma(-\psi(t_j) + 2)} \int_{\alpha}^{t_j} \frac{ds}{(-s + t_j)^{-1+\psi(t_j)}} - P_{i,2}(t_j)b_1(t_j) - P_{i,1}(t_j)b_2(t_j) - b_3(t_j),$$

so

$$G(i, j) = \frac{1}{\Gamma(-\psi(t_j) + 2)} \frac{(t_j - \alpha)^{-\psi(t_j)+2}}{-\psi(t_j) + 2} - P_{i,2}(t_j)b_1(t_j) - P_{i,1}(t_j)b_2(t_j) - b_3(t_j).$$

Case-3 For $t_j \in [\beta, \gamma)$ Eq. (15) becomes

$$G(i, j) = \frac{1}{\Gamma(-\psi(t_j) + 2)} \left(\int_0^{\alpha} \frac{h_i(s)ds}{(-s + t_j)^{-1+\psi(t_j)}} + \int_{\alpha}^{\beta} \frac{h_i(s)ds}{(-s + t_j)^{-1+\psi(t_j)}} + \int_{\beta}^{t_j} \frac{h_i(s)ds}{(-s + t_j)^{-1+\psi(t_j)}} \right) - P_{i,2}(t_j)b_1(t_j) - P_{i,1}(t_j)b_2(t_j) + b_3(t_j),$$

using value of $h_i(t_j)$ and simplifying, we obtain

$$G(i, j) = \frac{1}{\Gamma(-\psi(t_j) + 2)} \left(\int_{\alpha}^{\beta} \frac{ds}{(-s + t_j)^{-1+\psi(t_j)}} + \int_{\beta}^{t_j} \frac{-ds}{(-s + t_j)^{-1+\psi(t_j)}} \right) - P_{i,2}(t_j)b_1(t_j) - P_{i,1}(t_j)b_2(t_j) + b_3(t_j),$$

after further simplification, we obtain the following expression

$$G(i,j)=\frac{1}{\Gamma(-\psi(t_j)+2)}\left(\frac{(t_j-\alpha)^{-\psi(t_j)+2}-(t_j-\beta)^{-\psi(t_j)+2}}{-\psi(t_j)+2}-\frac{(t_j-\beta)^{-\psi(t_j)+2}}{-\psi(t_j)+2}\right)-P_{i,2}(t_j)b_1(t_j)-P_{i,1}(t_j)b_2(t_j)+b_3(t_j),$$

thus

$$G(i,j)=\frac{1}{\Gamma(-\psi(t_j)+2)}\left(\frac{(t_j-\alpha)^{-\psi(t_j)+2}}{-\psi(t_j)+2}-\frac{2(t_j-\beta)^{-\psi(t_j)+2}}{-\psi(t_j)+2}\right)-P_{i,2}(t_j)b_1(t_j)-P_{i,1}(t_j)b_2(t_j)+b_3(t_j).$$

Case-4 Now for $t_j\in[\gamma,1)$ Eq. (15) becomes

$$G(i,j)=\frac{1}{\Gamma(-\psi(t_j)+2)}\left(\int_0^{\alpha}\frac{h_i(s)ds}{(-s+t_j)^{-1+\psi(t_j)}}+\int_{\alpha}^{\beta}\frac{h_i(s)ds}{(-s+t_j)^{-1+\psi(t_j)}}+\int_{\beta}^{\gamma}\frac{h_i(s)ds}{(-s+t_j)^{-1+\psi(t_j)}}+\int_{\gamma}^{t_j}\frac{h_i(s)ds}{(-s+t_j)^{-1+\psi(t_j)}}\right)-P_{i,2}(t_j)b_1(t_j)-P_{i,1}(t_j)b_2(t_j),$$

using value of $h_i(t_j)$ and simplifying, we obtain

$$G(i,j)=\frac{1}{\Gamma(-\psi(t_j)+2)}\left(\int_{\alpha}^{\beta}\frac{ds}{(-s+t_j)^{-1+\psi(t_j)}}+\int_{\beta}^{\gamma}\frac{-ds}{(-s+t_j)^{-1+\psi(t_j)}}\right)-P_{i,2}(t_j)b_1(t_j)-P_{i,1}(t_j)b_2(t_j),$$

after further simplification, we obtain the following expression

$$G(i,j)=\frac{1}{\Gamma(-\psi(t_j)+2)}\left(\frac{(t_j-\alpha)^{-\psi(t_j)+2}}{-\psi(t_j)+2}-\frac{(t_j-\beta)^{-\psi(t_j)+2}}{-\psi(t_j)+2}-\frac{(t_j-\beta)^{-\psi(t_j)+2}}{-\psi(t_j)+2}+\frac{(t_j-\gamma)^{-\psi(t_j)+2}}{-\psi(t_j)+2}\right)-P_{i,2}(t_j)b_1(t_j)-P_{i,1}(t_j)b_2(t_j),$$

$$G(i,j)=\frac{1}{\Gamma(-\psi(t_j)+2)}\left(\frac{(t_j-\alpha)^{-\psi(t_j)+2}}{-\psi(t_j)+2}-\frac{2(t_j-\beta)^{-\psi(t_j)+2}}{-\psi(t_j)+2}+\frac{(t_j-\gamma)^{-\psi(t_j)+2}}{-\psi(t_j)+2}\right)-P_{i,2}(t_j)b_1(t_j)-P_{i,1}(t_j)b_2(t_j).$$

Thus, the values of matrix $G(i,j)$ is given by

$$G(i,j)=\begin{cases}0, & \text{if } t_j<0,\\ \frac{1}{\Gamma(-\psi(t_j)+2)}\frac{(t_j-\alpha)^{(-\psi(t_j)+2)}}{(-\psi(t_j)+2)}-P_{i,2}(t_j)b_1(t_j)-P_{i,1}(t_j)b_2(t_j)-b_3(t_j), & \text{if } t_j\in[\alpha,\beta),\\ \frac{1}{\Gamma(-\psi(t_j)+2)}\left(\frac{(t_j-\alpha)^{(-\psi(t_j)+2)}}{(-\psi(t_j)+2)}-\frac{2(t_j-\beta)^{(-\psi(t_j)+2)}}{(-\psi(t_j)+2)}\right)-P_{i,2}(t_j)b_1(t_j)-P_{i,1}(t_j)b_2(t_j) & \\ +b_3(t_j), & \text{if } t_j\in[\beta,\gamma),\\ \frac{1}{\Gamma(-\psi(t_j)+2)}\left(\frac{(t_j-\alpha)^{(-\psi(t_j)+2)}}{(-\psi(t_j)+2)}-\frac{2(t_j-\beta)^{(-\psi(t_j)+2)}}{(-\psi(t_j)+2)}+\frac{(t_j-\gamma)^{(-\psi(t_j)+2)}}{(-\psi(t_j)+2)}\right) & \\ -P_{i,2}(t_j)b_1(t_j)-P_{i,1}(t_j)b_2(t_j). & \text{if } t_j\in[\gamma,1).\end{cases}$$

Solving this system by Guass elimination scheme we obtain a_i's coefficients of Haar for $i=1,2,3...,N$. Using a_i's in Eq. (11) we obtain the required solution of Eq. (1).

Table 1: MAE, $R_c(N)$, RMSE and CPU Time (Seconds) for Example 1

J	$N = 2^{J+1}$	MAE	$R_c(N)$	RMSE	$R_c(N)$	CPU Time (Seconds)
1	4	1.37945×10^{-02}	—	9.91372×10^{-03}	—	0.015839
2	8	3.66734×10^{-03}	1.91128	2.45079×10^{-03}	2.01617	0.007070
3	16	9.46205×10^{-04}	1.95451	6.05229×10^{-04}	2.01769	0.013189
4	32	2.40325×10^{-04}	1.97716	1.49337×10^{-04}	2.01890	0.034763
5	64	6.05576×10^{-05}	1.98860	3.68279×10^{-05}	2.01970	0.096011
6	128	1.51990×10^{-05}	1.99432	9.08018×10^{-06}	2.02000	0.291041
7	256	3.80723×10^{-06}	1.99716	2.23934×10^{-06}	2.01964	1.024963
8	512	9.52742×10^{-07}	1.99858	5.52779×10^{-07}	2.01830	4.146401
9	1024	2.38302×10^{-07}	1.99929	1.36713×10^{-07}	2.01554	6.227488

4. Test Problems

Some of the examples in this section are solved using HWCM. To show the effectiveness the results are compared to exact solutions. Results are shown in tables and figures for each example. Maximum absolute error (MAE) is defined as

$$MAE = \max |y_{exa}(t) - y_{appr}(t)|,$$

if $y_exa(t)$ signifies the precise numerical solution and $y_appr(t)$ is approximate numerical solution at CPs and RMSE mean square root error at CPs is

$$RMSE = \sqrt{\frac{1}{N}\left(\sum_{i=1}^{N} |y_{exa}(t) - y_{appr}(t)|^2\right)}.$$

Problem 1. *Consider the following FDE of variable order*

$$D^{\psi(t)}y(t) + y'(t) = 1 + 2t + \frac{4t^{1-\psi(t)}(8+7t)}{(t-8)(t-4)\Gamma(-\psi(t)+1)}, \quad for \quad t \in [0,1], \tag{16}$$

where IC is $y(0) = 0$*. For* $\psi(t) = \frac{t}{4}$ *the exact solution of Eq.* (16) *is* $y(t) = t + t^2$.

Problem 2. *Consider the following FDE of variable order*

$$D^{\psi(t)}y(t) = \frac{18t^{2-\psi(t)}}{(18 - 9t + t^2)\Gamma(-\psi(t)+1)}, \quad for \quad t \in [0,1], \tag{17}$$

where IC is $y(0) = 0$*. For* $\psi(t) = \frac{t}{3}$ *the exact solution of Eq.* (17) *is* $y(t) = t^2$.

Problem 3. *Consider the following FDE of variable order*

$$D^{\psi(t)}y(t) = 5\left(\frac{2}{\Gamma(3-\psi(t))}t^{2-\psi(t)} + \frac{3}{\Gamma(-\psi(t)+2)}t^{1-\psi(t)}\right), \quad for \quad t \in [0,1], \tag{18}$$

where IC is $y(0) = 0$*. For* $\psi(t) = \frac{3}{5}(\cos t + \sin t)$ *the exact solution of Eq.* (18) *is* $y(t) = 5(3t + t^2)$.

Problem 4. *Consider the following FDE of variable order*

$$D^{\psi(t)}y(t) = \frac{3t^{1-\psi(t)}}{\Gamma(-\psi(t)+2)} + \frac{2t^{2-\psi(t)}}{\Gamma(3-\psi(t))}, \quad for \quad t \in [0,1], \tag{19}$$

where IC is $y(0) = 0$*. For* $\psi(t) = \frac{t}{2}$ *the exact solution of Eq.* (19) *is* $y(t) = t^2 + 3t$.

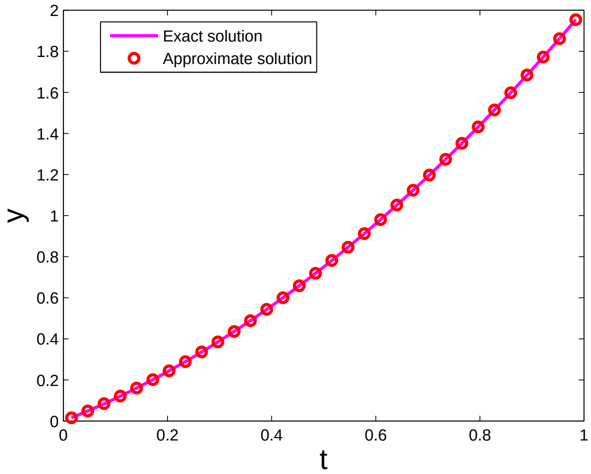

Figure 1: Comparison of exact and approximate solution for 32 CPs of Example 1

Table 2: MAE, $R_c(N)$, RMSE and CPU Time (Seconds) for Example 2

J	$N = 2^{J+1}$	MAE	$R_c(N)$	RMSE	$R_c(N)$	CPU Time (Seconds)
1	4	7.52783×10^{-03}	—	4.26440×10^{-03}	—	0.001870
2	8	2.27310×10^{-03}	1.72757	1.23892×10^{-03}	1.78325	0.002061
3	16	7.63220×10^{-04}	1.57448	3.94087×10^{-04}	1.65249	0.011010
4	32	2.54310×10^{-04}	1.58550	1.25102×10^{-04}	1.65540	0.131003
5	64	8.27591×10^{-05}	1.61960	3.92467×10^{-05}	1.67246	0.188925
6	128	2.65215×10^{-05}	1.64175	1.21911×10^{-05}	1.68674	0.186648
7	256	8.41452×10^{-06}	1.65620	3.75883×10^{-06}	1.69747	0.617863
8	512	2.65177×10^{-06}	1.66592	1.15264×10^{-06}	1.70534	2.331360
9	1024	8.31829×10^{-07}	1.67260	3.52063×10^{-07}	1.71103	9.583538

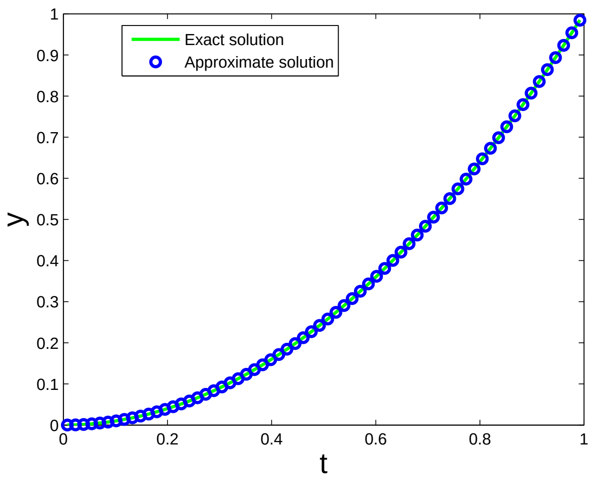

Figure 2: Comparison of exact and approximate solution for 64 CPs of Example 2

Table 3: MAE, $R_c(N)$, RMSE and CPU Time (Seconds) for Example 3

J	$N = 2^{J+1}$	MAE	$R_c(N)$	RMSE	$R_c(N)$	CPU Time (Seconds)
1	4	5.16684×10^{-02}	—	3.67380×10^{-02}	—	0.000878
2	8	3.32274×10^{-02}	0.63690	2.06976×10^{-02}	0.82780	0.001506
3	16	1.67025×10^{-02}	0.99230	1.03296×10^{-02}	1.00267	0.005404
4	32	7.74384×10^{-03}	1.10895	4.75119×10^{-03}	1.12042	0.019453
5	64	3.47063×10^{-03}	1.15784	2.10422×10^{-03}	1.17499	0.082697
6	128	1.53188×10^{-03}	1.17989	9.15799×10^{-04}	1.20018	0.278190
7	256	6.71554×10^{-04}	1.18972	3.95499×10^{-04}	1.21135	1.062427
8	512	2.93602×10^{-04}	1.19363	1.70305×10^{-04}	1.21554	4.119492
9	1024	1.28278×10^{-04}	1.19458	7.33031×10^{-05}	1.21618	6.318862

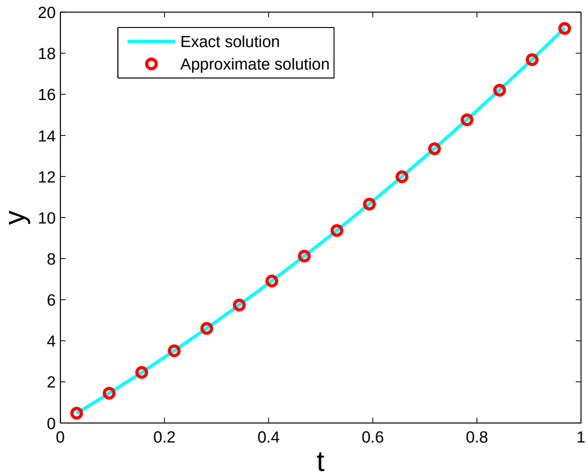

Figure 3: Comparison of exact and approximate solution for 16 CPs of Example 3

Table 4: MAE, $R_c(N)$, RMSE and CPU Time (Seconds) for Example 4

J	$N = 2^{J+1}$	MAE	$R_c(N)$	RMSE	$R_c(N)$	CPU Time (Seconds)
1	4	1.00221×10^{-02}	—	5.87807×10^{-03}	—	0.001499
2	8	3.52203×10^{-03}	1.50870	1.92366×10^{-03}	1.61148	0.003960
3	16	1.30121×10^{-03}	1.43655	6.55237×10^{-04}	1.55376	0.012229
4	32	4.65935×10^{-04}	1.48165	2.20884×10^{-04}	1.56872	0.048928
5	64	1.63360×10^{-04}	1.51206	7.37026×10^{-05}	1.58350	0.287627
6	128	5.65788×10^{-05}	1.52972	2.44278×10^{-05}	1.59319	0.341924
7	256	1.94591×10^{-05}	1.53981	8.06534×10^{-06}	1.59871	0.935699
8	512	6.66712×10^{-06}	1.54531	2.65824×10^{-06}	1.60126	3.879564
9	1024	2.28007×10^{-06}	1.54798	8.75850×10^{-07}	1.60171	6.549134

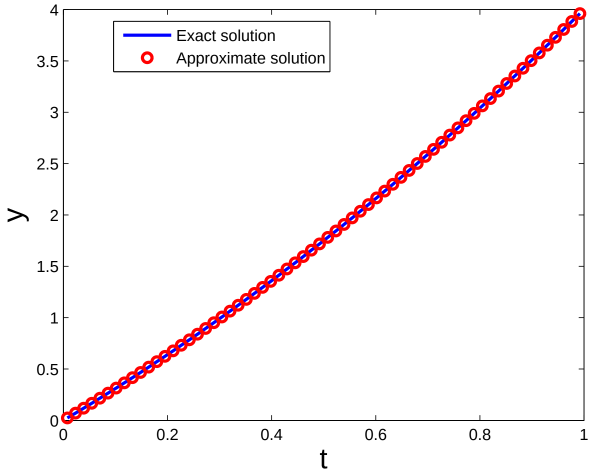

Figure 4: Comparison of 64 CPs exact and approximative solutions 4

5. Results and discussion

Tables are provided with the MAE and RMSE for each case for various numbers of CPs. Tables show that as the number of CPs is increased, both the MAE and RMSE errors decrease. Additionally, for each of the samples, we estimated the CPU time (seconds). We also calculated $R_c(N)$ for each example and found from tables that it is almost equal to 2, supporting the theoretical findings of Majak et al. [26]. For each case, we also included a figure comparing the exact and approximative solutions for various numbers of collocation points. Figure makes it obvious that the exact and approximative solutions are close to one another. In MATLAB programme, we completed all of the computational work. Thus, we draw the conclusion that the HWCM is a useful technique for finding the numerical solution of FDEs with linear variable orders.

6. Conclusion

n HWCM is developed for the IC based solution of linear VO FDEs. The accuracy and effectiveness of HWCM are evaluated using the numerical approach on a variety of examples. The results of each case for different quantities of CPs are shown in tables. Figures compare exact and approximative solutions for various numbers of CPs. We may therefore draw the conclusion that the HWCM is an effective method for finding the solution of linear VO FDEs.

Availability of data

The paper includes the data that were utilized in this study.

Conflict of interest

No conflict of interest exist.

Authors contribution:

Each author has contributed equally to the work.

Acknowledgment

The authors extend their appreciation to the Deanship of Scien- tific Research at King Khalid University, Saudi Arabia for funding this work through large group Research Project under Grant number RGP2/371/44.

Funding

No external funding received.

References

[1] I. Podlubny, *Fractional differential equations*, Academic press, New York, 1999.
[2] L. Liu, J. Wang, L. Zhang, S. Zhang, *Multi-AUV dynamic maneuver countermeasure algorithm based on interval information game and fractional-order DE*, Fractal Fract., 6(5) (2022), 235.
[3] H. Y. Jin, Z. A. Wang, *Global stabilization of the full attraction-repulsion Keller-Segel system*, Discrete and Continuous Dyn. Sys. Series A, 40(6), (2022), 3509–3527.
[4] V. N. Kovalnogov, R. V. Fedorov, Y. A. Khakhalev, T. E. Simos, C. Tsitouras, *A neural network technique for the derivation of Runge-Kutta Pairs adjusted for scalar autonomous problems*, Math., 9 (2021), 1842.
[5] Y. Tang, S. Liu, Y. Deng, Y. Zhang, L. Yin, W. Zheng, *An improved method for soft tissue modeling*, Biomedical signal processing and control, 65 (2021).
[6] W. Zheng, X. Liu, X. Ni, L. Yin, B. Yang, *Improving visual reasoning through semantic representation*, IEEE access, 9 (2021), 91476–91486.
[7] W. Zheng, X. Liu, L. Yin, *Sentence representation method based on multi-layer semantic network*, Appl. Sci., 11(3) (2021), 1316.
[8] H. Jin, Z. Wang, L. Wu, *Global dynamics of a three-species spatial food chain model*, J. Diff. Equ., 333 (2022), 144–183.
[9] F. Ghoreishi, S. Yazdani, *An extension of the spectral Tau method for the numerical solution of multi-order fractional differential equations with convergence analysis*, Comput. Math. Appl., 61 (2011), 30-43.
[10] E. H. Doha, A. H. Bhrawy, D. Baleanu, S. S. Ezz-Eldien, *On shifted Jacobi spectral approximations for solving fractional differential equations*, Appl. Math. comput., 219 (2013), 8042-8056.
[11] S. Esmaeili, M. Shamsi, Y. Luchko, *Numerical solution of fractional differential equations with a collocation method based on Müntz polynomials*, Comput. Math. Appl., 62 (2011), 918-929.
[12] K. Diethelm, N. J. Ford, A. D. Freed, *A predicator-corrector approach for the numerical solution of fractional differential equations*, Nonlinear Dyn., 29 (2002), 3-22.
[13] R. B. Albadarneh, I. M. Batiha, M. Zurigat, *Numerical solutions for linear fractional differential equations of order* $1 < \alpha < 2$ *using finite difference method (FFDM)*, J. Math. comput. Sci., 16 (2016), 103-111.
[14] Y. Li, *Solving a nonlinear fractional differential equation using Chebyshev wavelets*, Commun. nonlinear Sci. Numer. Simul., 15 (2010), 2284-2292.
[15] A. Arikoglu, I. Ozkol, *Solution of fractional differential equations by using differential transform method*, Chaos, solitons and fractals, 34 (2007), 1473-1481.
[16] S. Momani, Z. Odibat, *Numerical comparison of methods for solving linear differential equations of fractional order*, Chaos, solitons and fractals, 31 (2007), 1248-1255.
[17] M. Faheem, A. Khan, A. Raza, *Wavelet collocation methods for solving neutral delay differential equations*, Int. J. Nonlinear Sci. Numer. Simul., 23(2022), 1129–1156.
[18] M. Faheem, A. Khan, A. Raza, *Collocation methods based on Gegenbauer and Bernoulli wavelets for solving neutral delay differential equations*, Math. Comput. Simul., 180 (2021), 72–92
[19] M. Faheem, A. Khan, A. Raza, *A high resolution Hermite wavelet technique for solving space time-fractional partial differential equations*, Math. Comput. Simul., 194 (2022), 588–609.
[20] U. Lepik, H. Hein, *Haar wavelets with applications*, springer, Switzerland, 2014.
[21] R. Amin, K. Shah, M. Asif, I. Khan, F. Ullah, *An efficient algorithm for numerical solution of fractional integro-differential equations via Haar wavelet*, J. Comput. Appl. Math., 381(2021), 113028.
[22] I. Aziz, R. Amin, *Numerical solution of a class of delay differential and delay partial differential equations via haar wavelet*, Appl. Math. Model., 40 (2016) 10286–10299.
[23] R. Amin, K. Shah, M. Asif, I. Khan, *A computational algorithm for the numerical solution of fractional order delay differential equations*, Appl. Math. Comput., 402 (2021), 125863.
[24] R. Amin, B. Alshahrani, A. H. Aty, K. Shah, W. Deebani, *Haar wavelet method for solution of distributed order time-fractional differential equations*, Alex. Eng. J., 60:3 (2021), 3295-3303.
[25] R. Amin, H. Ahmad, K. Shah, M. B. Hafeez, W. Sumelka, *Theoretical and computational analysis of nonlinear fractional integro-differential equations via collocation method*, Chaos, Solitons and Fractals, 151 (2021) 111252.
[26] J. Majak, B. S. Shvartsman, M. Kirs, M. Pohlak, H. Herranen, *Convergence theorem for the Haar wavelet based discretization method*, Compos. Struct., 126 (2015), 227-232.