Few-Body Syst (2017) 58:156
https://doi.org/10.1007/s00601-017-1319-x

Harendra Singh · Rajesh K. Pandey · D. Baleanu

Stable Numerical Approach for Fractional Delay Differential Equations

Received: 4 January 2017 / Accepted: 13 September 2017 / Published online: 20 September 2017
© Springer-Verlag GmbH Austria 2017

Abstract In this paper, we present a new stable numerical approach based on the operational matrix of integration of Jacobi polynomials for solving fractional delay differential equations (FDDEs). The operational matrix approach converts the FDDE into a system of linear equations, and hence the numerical solution is obtained by solving the linear system. The error analysis of the proposed method is also established. Further, a comparative study of the approximate solutions is provided for the test examples of the FDDE by varying the values of the parameters in the Jacobi polynomials. As in special case, the Jacobi polynomials reduce to the well-known polynomials such as (1) Legendre polynomial, (2) Chebyshev polynomial of second kind, (3) Chebyshev polynomial of third and (4) Chebyshev polynomial of fourth kind respectively. Maximum absolute error and root mean square error are calculated for the illustrated examples and presented in form of tables for the comparison purpose. Numerical stability of the presented method with respect to all four kind of polynomials are discussed. Further, the obtained numerical results are compared with some known methods from the literature and it is observed that obtained results from the proposed method is better than these methods.

1 Introduction

The standard linear delay differential equation (DDE) is given by

$$y'(x) = f\left(x, y(x), y'(x), y(cx-\tau(x)), y'(cx-\tau(x))\right), \tag{1}$$

where C is a constant and $\tau(x)$ is delay and $cx-\tau(x)$ is delay argument. $\tau(x)$ is called constant or time dependent delay depending on τ is constant or function of time variable x respectively. DDEs have many applications in the field of life sciences [1] and in various mathematical model of biological systems for example population dynamics and infectious diseases models [2], time-delay model of single-species growth with stage structure [3], two- point bour problem in viscoelastic flows model [4], stage structured predator–prey model [5], glucose-insulin regulatory system and ultradian insulin secretory oscillations model [6].

H. Singh · R. K. Pandey (✉) · D. Baleanu
Department of Mathematical Sciences, Indian Institute of Technology, Banaras Hindu University, Varanasi 221005, India
E-mail: rkpandey.mat@iitbhu.ac.in

H. Singh
E-mail: harendra059@gmail.com

D. Baleanu
Department of Mathematics, Cankaya University, Ankara, Turkey

D. Baleanu
Institute of Space Sciences, Măgurele, Bucharest, Romania

Due to non-local nature of fractional derivative it is physically important to study fractional differential equations. Some physical quantity depends on the past so fractional model of such differential equation becomes important to better understand the physical model. Fractional calculus has many real applications in science and engineering such as biology [7], viscoelasticity [8–10], signal processing [11], bioengineering [12] and control theory [13].

In this paper, we consider a more general form of FDDE by replacing integer order time derivative by fractional order Caputo derivative as follows:

$$D_c^{\alpha} y(x) = f\left(x, y(x), D_c^{\beta} y(x), y(cx - \tau(x)), D_c^{\gamma} y(cx - \tau(x))\right), \quad \alpha \geq \beta, \gamma > 0. \tag{2}$$

Recently, many authors solved FDDEs using numerous numerical techniques. In [14], authors used steps method to solve DDEs of integer order and in [1] similar technique is used to solve FDDEs by converting it into non-delay differential equation. Lv and Gao [15] used the reproducing kernel Hilbert space method (RKHSM) for solving neutral functional differential equations with proportional delays. In [16], Wang used the Runge–Kutta type method to obtain approximate solution of FDDE and discussed stability of the method. Some researchers used wavelets to solve FDDEs. In [17,18], Hermite wavelets and Chebyshev wavelets method are used to obtain approximate solution of FDDE. Some more papers for approximate solution of FDDE are given in [19–24].

In present paper, we are using operation matrix of Jacobi polynomial to solve more general form of FDDE. In this method by taking finite dimensional approximation of unknown function and using operational matrix of integration in the FDDE, we obtain a set of linear algebraic equations whose solution gives approximate solution of the FDDE. Stability of the proposed scheme is discussed numerically. A comparative study of the presented scheme using different polynomials is presented for all the test examples. Numerical results are discussed for each example varying the fractional order α, time delay τ and compared with the known analytical solution.

2 Preliminaries

The fractional order differentiation and integration are defined as follows,

Definition 2.1 The Riemann–Liouville fractional order integral operator is given by

$$J^{\alpha} g(y) = \frac{1}{\Gamma\alpha} \int_0^y (y-t)^{\alpha-1} g(t) dt \quad \alpha > 0, y > 0,$$
$$J^0 g(y) = g(y).$$

For the fractional Riemann–Liouville integration

$$J^{\alpha} y^k = \frac{\Gamma(k+1)}{\Gamma(k+1+\alpha)} y^{k+\alpha}$$

Definition 2.2 The Caputo fractional derivative of order β are defined as

$$D_c^{\beta} g(y) = J^{l-\beta} D_c^l g(y) = \frac{1}{\Gamma(l-\beta)} \int_0^y (y-t)^{l-\beta-1} \frac{d^l}{dt^l} g(t) dt,$$
$$l - 1 < \beta < l, y > 0.$$

In present paper we use the following property of fractional Caputo derivative,

$$D_c^{\beta} A = 0 \quad \text{(A is a constant)},$$
$$D_c^{\beta} y^k = \begin{cases} \frac{\Gamma(k+1)}{\Gamma(k+1-\beta)} y^{k-\beta}, & \text{for } k \in \mathrm{N}_0 \text{ and } k \geq \lceil \beta \rceil, \\ 0, & \text{for } k \in \mathrm{N}_0 \text{ and } k < \lceil \beta \rceil, \end{cases}$$

where symbols have their usual meanings, while $\mathrm{N} = \{1, 2, 3, \ldots\}$ and $\mathrm{N}_0 = \{0, 1, 2, \ldots\}$.

The shifted Jacobi polynomial of degree i on [0, 1] is given as,

$$\mu_i^{(u,v)}(y) = \sum_{k=0}^{i} (-1)^{i-k} \frac{\Gamma(i+v+1)\,\Gamma(i+k+u+v+1)}{\Gamma(k+v+1)\,\Gamma(i+u+v+1)\,(i-k)!k!} y^k \quad . \tag{3}$$

and satisfies the following relation:

$$\int_0^1 \mu_n^{(u,v)}(y)\mu_m^{(u,v)}(y)w^{(u,v)}(y)dy = l_n^{u,v}\delta_{nm}, \tag{4}$$

where δ_{nm} is Kronecker delta function and,

$$l_n^{u,v}\delta_{nm} = \frac{\Gamma(n+u+1)\,\Gamma(n+v+1)}{(2n+u+v+1)n!\,\Gamma(n+u+v+1)} \quad . \tag{5}$$

A function $g \in L^2[0,1]$, with bounded second derivative $\left|g''(y)\right| \le K$, can be expanded as,

$$g(y) = \lim_{n\to\infty} \sum\nolimits_{k=0}^{n} c_k \mu_k^{(u,v)}(y), \tag{6}$$

where,

$$c_k = \frac{1}{l_k^{u,v}} \int\limits_0^1 \mu_k^{(u,v)}(y)g(y)w^{(u,v)}(y)dy, \quad k = 0, 1, 2, \ldots \tag{7}$$

If the series is truncated at $n = m$, then we have,

$$g \cong \sum\nolimits_{k=0}^{m} c_k \mu_k^{(u,v)}(y) = C^T \phi_m(y), \tag{8}$$

where C and $\phi_m(y)$ are $(m+1)\times 1$ matrices given by

$$C = [c_0, c_1, \ldots, c_m]^T \text{ and } \phi_m(y) = [\mu_0^{(u,v)}(y), \mu_1^{(u,v)}(y), \ldots, \mu_m^{(u,v)}(y)]^T. \tag{9}$$

3 Operational Matrix

Theorem 3.1 *Let* $\phi_m(y) = [\mu_0^{(u,v)}(y), \mu_1^{(u,v)}(y), \ldots, \mu_m^{(u,v)}(y)]^T$, *be the column vector consisting of the shifted Jacobi polynomials and consider* $\alpha > 0$, *then*

$$J^\alpha \phi_m(y) = J^{(\alpha)}\phi_m(y), \tag{10}$$

where, $J^{(\alpha)} = (\omega(i,j))$, *is* $(m+1)\times(m+1)$ *operational matrix of fractional integral of order* α *and its entries are given by*

$$\omega(i,j,u,v) = \sum_{k=0}^{i} (-1)^{i-k} \frac{\Gamma(i+v+1)\,\Gamma(i+k+u+v+1))}{\Gamma(k+v+1)\,\Gamma(i+u+v+1)\,(i-k)!\,\Gamma(k+\alpha+1)}$$
$$\times \sum_{l=0}^{j} (-1)^{j-l} \frac{\Gamma(j+l+u+v+1)\,\Gamma(u+1)\,\Gamma(k+l+\alpha+v+1)\,(2j+u+v+1)j!}{\Gamma(j+u+1)\,\Gamma(l+v+1)\,(j-l)!(l)!\,\Gamma(l+k+u+v+\alpha+2)},$$
$$0 \le i \le m, 0 \le j \le m. \tag{11}$$

Proof Pl. see [25–27].

4 Method of Solution

The general form of the FDDEs is stated as

$$D_c^{\alpha} y(x) = k_1 y(x) + k_2 D_c^{\beta} y(x) + k_3 y(cx - \tau(x)) + k_4 D_c^{\gamma} y(cx - \tau(x)) + k_5 D_c^{\delta} u(x), \tag{12}$$

where $\alpha \geq \beta, \gamma > 0$, with initial conditions $y^l(0) = d_l, l = 0, 1, 2, \ldots, k-1; k-1 < \alpha \leq k$; $k_1, k_2, k_3, k_4, k_5, c$ are real constants, D_c^{α}, D_c^{β}, D_c^{γ} are Caputo derivative of order α, β and γ respectively, $\tau(x)$ and $u(x)$ are delay and any known function respectively.

For solving Eq. (12), we take the following approximations,

$$D_c^{\alpha} y(x) = C^T \phi_n(x), \tag{13}$$

$$h(x) = D_c^{\delta} u(x) \approx B^T \phi_n(x), \tag{14}$$

where C and $\phi_n(x)$ are $(n+1) \times 1$ matrices given by

$$C = [c_0, c_1, \ldots, c_n]^T \text{ and } \phi_n(x) = [\mu_0^{(u,v)}(x), \mu_1^{(u,v)}(x), \ldots, \mu_n^{(u,v)}(x)]^T.$$

Taking integral of order α on both side of Eq. (13), we get

$$y(x) = C^T J^{(\alpha)} \phi_n(x) + A^T \phi_n(x), \tag{15}$$

where, $\sum_{l=0}^{k-1} y^{(l)}(0) \frac{x^l}{l!} = A^T \phi_n(x)$ and $J^{(\alpha)}$ is $(n+1) \times (n+1)$ operational matrix of integrations.

Using Eq. (15), approximation for the delay function is given as

$$y(cx - \tau(x)) = C^T J^{(\alpha)} \phi_n(cx - \tau(x)) + A^T \phi_n(cx - \tau(x)), \tag{16}$$

From Eq. (13), we can write,

$$D_c^{\beta} y(x) = J^{\alpha-\beta} D_c^{\alpha} y(x) \approx J^{\alpha-\beta} \left(C^T \phi_n(x) \right) = C^T J^{\alpha-\beta} \phi_n(x) \approx C^T J^{(\alpha-\beta)} \phi_n(x), \tag{17}$$

From Eq. (17), we can write the following approximation,

$$D_c^{\gamma} y(cx - \tau(x)) \approx C^T J^{(\alpha-\gamma)} \phi_n(cx - \tau(x)), \tag{18}$$

Using Eqs. (13)–(18) in Eq. (12), we get

$$\begin{aligned} C^T \phi_n(x) = {} & k_1(C^T J^{(\alpha)} + A^T)\phi_n(x) + k_2 C^T J^{(\alpha-\beta)} \phi_n(x) + k_3(C^T J^{(\alpha)} + A^T)\phi_n(cx - \tau(x)) \\ & + k_4 C^T J^{(\alpha-\gamma)} \phi_n(cx - \tau(x)) + k_5 B^T \phi_n(x), \end{aligned} \tag{19}$$

Further,

$$\phi_n(cx - \tau(x)) \approx E^T \phi_n(x), \text{ where } E \text{ is a square matrix of order } (n+1) \times (n+1). \tag{20}$$

Now from Eqs. (19) and (20), we can write

$$C^T \phi_n(x) = \left(k_1(C^T J^{(\alpha)} + A^T) + k_2 C^T J^{(\alpha-\beta)} + k_3(C^T J^{(\alpha)} + A^T)E^T + k_4 C^T J^{(\alpha-\gamma)} E^T + k_5 B^T \right) \phi_n(x),$$

from the above equation, we can write,

$$C^T = \left(k_1 A^T + k_3 A^T E^T + k_5 B^T \right) \left(I - k_1 J^{(\alpha)} - k_2 J^{(\alpha-\beta)} - k_3 J^{(\alpha)} E^T - k_4 J^{(\alpha-\gamma)} E^T \right)^{-1}, \tag{21}$$

where I is an identity matrix.

Using the value of C^T from Eq. (21) in (15), we get approximate solution for FDDEs given in Eq. (12).

5 Error Analysis

Lemma 1 *Let $y \in C^{(n+1)}[0,1]$ with bounded derivatives of all order and $S_n = Span\{\mu_0^{(u,v)}(x), \mu_1^{(u,v)}(x), \ldots, \mu_n^{(u,v)}(x)\}$. If y_n be the n^{th} approximation of y then we have followings inequality,*

$$\|y - y_n\|_{L^2[0,1]} \le \frac{K}{(n+1)!}\sqrt{B(u+1, v+1)}, \tag{22}$$

where $K = \max_{x \in [0,1]} \left| y^{(n+1)}(x) \right|$.

Proof see [28].

Theorem *Let $(J_n^\alpha g)(y)$ be the n^{th} approximation of Riemann–Liouville fractional integral operator $(J^\alpha g)(y)$ then we have the followings upper bounds of absolute error in its n^{th} approximation,*

$$\|g(y) - g_n(y)\|_{L^2[0,1]} \le \frac{K}{\Gamma(\alpha+1)\ (n+1)!}\sqrt{B(u+1, v+1)}. \tag{23}$$

Proof By the definition of fractional integral operator, we can write

$$\left\|(J^\alpha g)(y) - (J_n^\alpha g)(y)\right\|_{L^2[0,1]} = \frac{1}{\Gamma\alpha}\int_0^y (y-\tau)^{\alpha-1}\,|g(\tau) - g_n(\tau)|d\tau, \tag{24}$$

Using Eq. (22) in (24), we get,

$$\left\|(J^\alpha g)(y) - (J_n^\alpha g)(y)\right\|_{L^2[0,1]} \le \frac{1}{\Gamma\alpha}\int_0^y (y-\tau)^{\alpha-1}\frac{K}{(n+1)!}\sqrt{B(u+1, v+1)}d\tau,$$

$$= \frac{K}{\Gamma\alpha\ (n+1)!}\sqrt{B(u+1, v+1)}\int_0^y (y-\tau)^{\alpha-1}d\tau,$$

$$\left\|(J^\alpha g)(y) - (J_n^\alpha g)(y)\right\|_{L^2[0,1]} \le \frac{K}{\Gamma(\alpha+1)\ (n+1)!}\sqrt{B(u+1, v+1)}y^\alpha. \tag{25}$$

Since $y \in [0, 1]$, Eq. (25) can be written as,

$$\left\|(J^\alpha g)(y) - (J_n^\alpha g)(y)\right\|_{L^2[0,1]} \le \frac{K}{\Gamma(\alpha+1)\ (n+1)!}\sqrt{B(u+1, v+1)}.$$

This completes the proof.

6 Numerical Stability

Here, we present the numerical stability in similar approach as described in [29]. To show the accuracy of proposed method we have calculated absolute error and root mean square error (RMS). These errors are given by the following equations,

$$\Delta y(x_i) = |y_e(x_i) - y_a(x_i)|\,, \tag{26}$$

and

$$\sigma_{(N+1)^2} = \left\{\frac{1}{(N+1)}\sum_{i=0}^{N}[y_e(x_i) - y_a(x_i)]^2\right\}^{1/2}, \tag{27}$$

where $y_e(x_i)$ and $y_a(x_i)$ are the exact and approximate value of output function at point x_i.

Table 1 Noise reduction $H(x)$ for $N = 10$ and $n = 5$ at different values of δ

x	$\delta = 0.01$	$\delta = 0.001$	$\delta = 0.0001$
0	0.0000	0.00000	0.000000
0.1	0.0020	0.00020	0.000020
0.2	0.0040	0.00040	0.000040
0.3	0.0060	0.00060	0.000060
0.4	0.0080	0.00080	0.000080
0.5	0.0100	0.00100	0.000100
0.6	0.0120	0.00120	0.000120
0.7	0.0140	0.00140	0.000140
0.8	0.0160	0.00160	0.000160
0.9	0.0180	0.00180	0.000180
1.0	0.0200	0.00200	0.000200

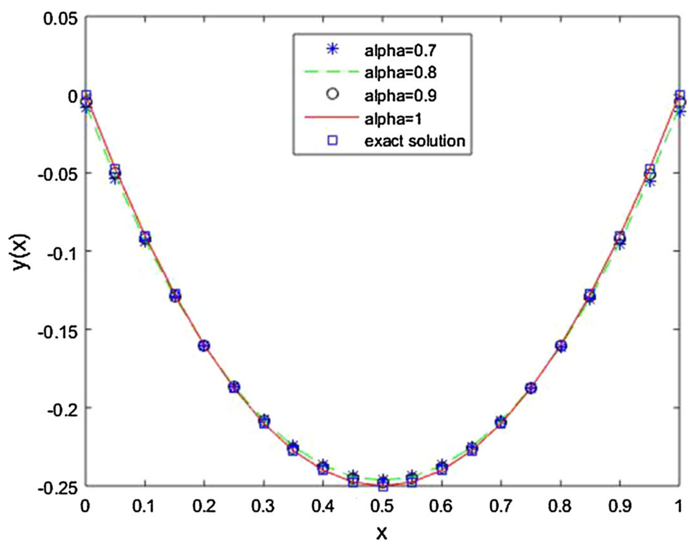

Fig. 1 The behaviour of solution for different values of α at u = 0, v = 0, Example 1

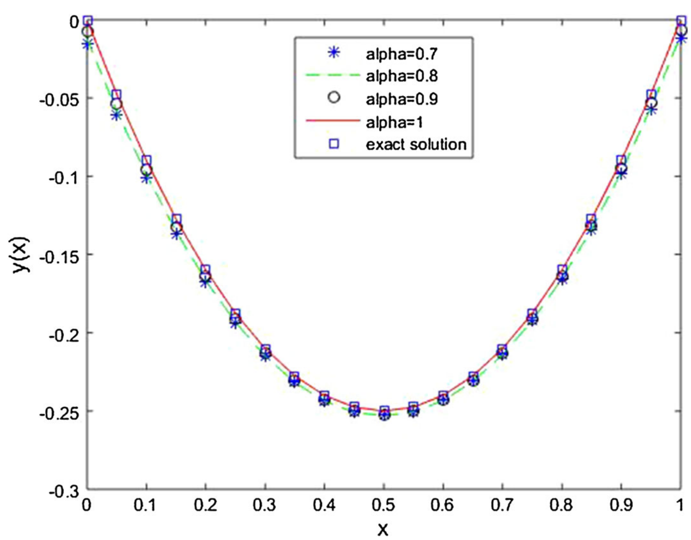

Fig. 2 The behaviour of solution for different values of α at u = 1/2, v = 1/2, Example 1

From now wards, considering $h(x)$ as input function. Adding random noise term in input function we demonstrate the stability of the proposed method.

In all examples, $h(x)$ and $h^{\delta}(x)$, represent the exact and input function with noise respectively. The noise function $h^{\delta}(x)$ is obtained from $h(x)$ by adding noise term, $h^{\delta}(x_i) = h(x_i) + \delta\theta_i$, where $x_i = ih, i = 1, 2, \ldots, N, Nh = 1$; and $\theta_i \in [0, 1]$, is the uniform random variable such that,

$$\max_{1 \le i \le N} \left| h^{\delta}(x_i) - h(x_i) \right| \le \delta. \tag{28}$$

Reconstructed output function $y_a^{\delta}(x)$ (with δ noise) and $y_a^0(x)$ (without noise) are obtained with and without noise term in the input function $h(x)$ and using Eq. (15) these are given by

$$y_a^{\delta}(x) \cong C^{\delta T} J^{(\alpha)} \phi_n(x) + A^T \phi_n(x), \tag{29}$$

$$y_a^0(x) \cong C^T J^{(\alpha)} \phi_n(x) + A^T \phi_n(x), \tag{30}$$

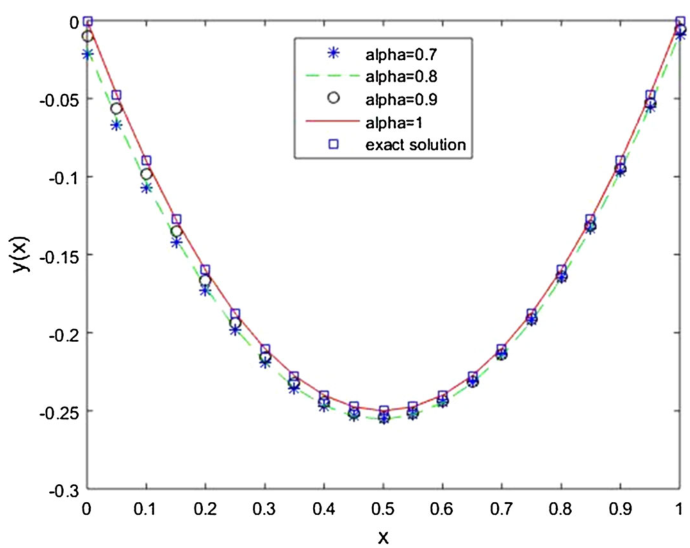

Fig. 3 The behaviour of solution for different values of α at u $= -1/2$, v $= 1/2$, Example 1

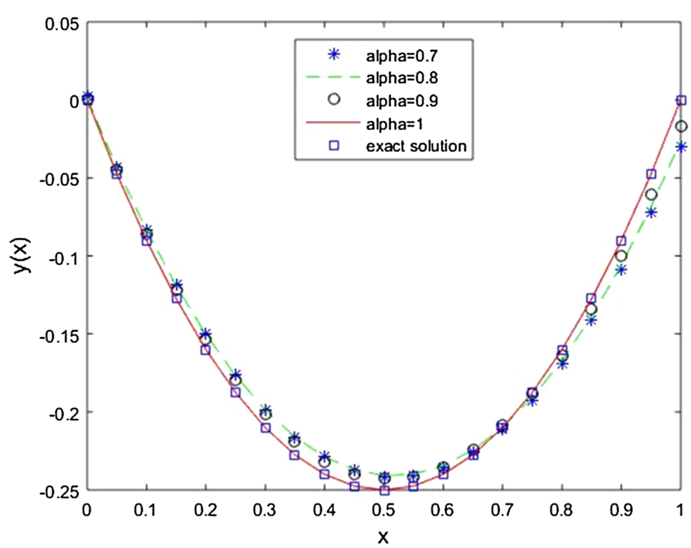

Fig. 4 The behaviour of solution for different values of α at u $= 1/2$, v $= -1/2$, Example 1

where $C^{\delta T}$ and C^T are known matrices and they are given by

$$C^{\delta T} = \left(k_1 A^T + k_3 A^T E^T + k_5 B^{\delta T}\right)\left(I - k_1 J^{(\alpha)} - k_2 J^{(\alpha-\beta)} - k_3 J^{(\alpha)} E^T - k_4 J^{(\alpha-\gamma)} E^T\right)^{-1}$$

and

$$C^T = \left(k_1 A^T + k_3 A^T E^T + k_5 B^T\right)\left(I - k_1 J^{(\alpha)} - k_2 J^{(\alpha-\beta)} - k_3 J^{(\alpha)} E^T - k_4 J^{(\alpha-\gamma)} E^T\right)^{-1},$$

we approximate $h^\delta(x)$ as

$$h^\delta(x) = h(x) + \delta\theta_i \cong B^\delta \phi_n(x). \tag{31}$$

From Eqs. (29) and (30)

$$y_a^\delta(x) - y_a^0(x) \cong \left(C^{\delta T} - C\right) J^{(\alpha)} \phi_n(x). \tag{32}$$

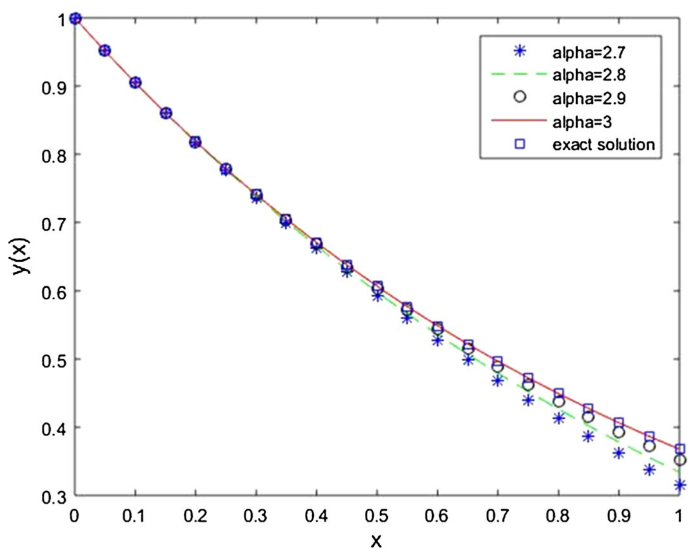

Fig. 5 The behaviour of solution for different values of α at u = 0, v = 0, Example 2

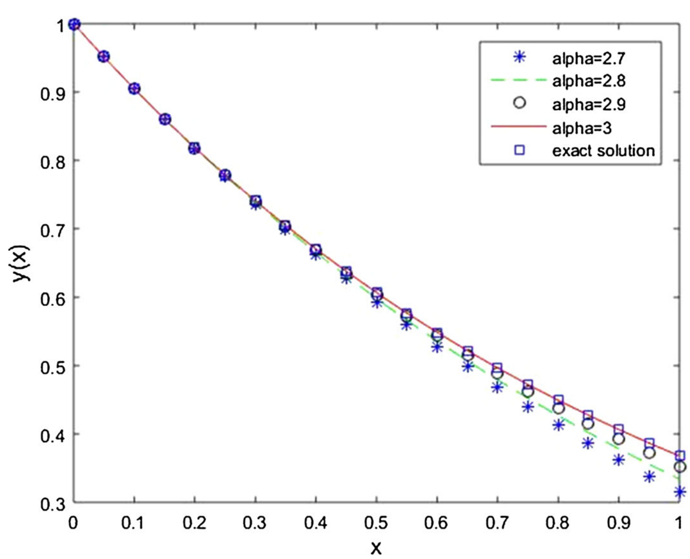

Fig. 6 The behaviour of solution for different values of α at u = 1/2, v = 1/2, Example 2

Let

$$H(x) = y_a^{\delta}(x) - y_a^{0}(x) \cong \left(C^{\delta T} - C^T\right) J^{(\alpha)} \phi_n(x), \tag{33}$$

then $H(x)$ reflects the noise reduction capability of the method. Its values at various points are shown in Table 1.

From Table 1, it is clear that there is a continuous variation in noise reduction capability which shows that our method is stable in [0, 1]. In Sect. 7, we illustrate the stability of the proposed method by calculating errors with and without noises. In last two examples, we add two different noises $\delta_1 = \sigma_{(N+1)^2}$ (root mean square error), $\delta_2 = E_\infty$ (maximum absolute error) for $N = 30$. We plot absolute errors graph without and with noises (δ_1, δ_2) taking $n = 7$, $N = 30$. Figs. 9, 10, 11, 12 and 17, 18, 19, 20 are plotted for Exp. 2 and 3 taking different cases of Jacobi polynomial and it is observed that the there is a very small change in absolute error when we add noises in input function showing the stability of our method.

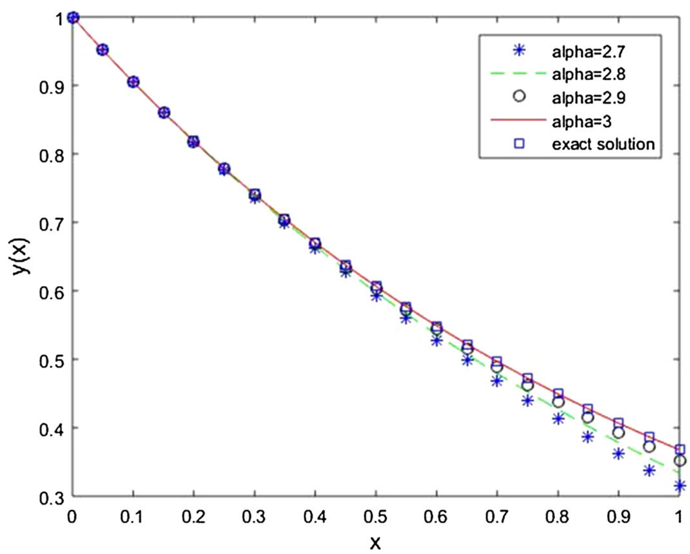

Fig. 7 The behaviour of solution for different values of α at u$= -1/2$, v$= 1/2$, Example 2

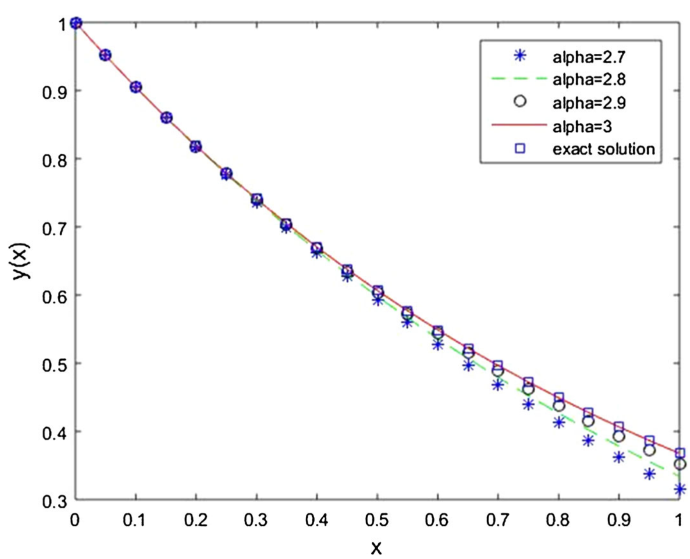

Fig. 8 The behaviour of solution for different values of α at u$= 1/2$, v$= -1/2$, Example 2

7 Numerical Results and Discussions

In present section, we examine the accuracy and numerical stability of the presented method by testing it on some numerical examples. We apply our algorithm on three test problem and compare the results from the existing methods. In graphical presentation of absolute errors with and without noise term, the noises δ_1 and δ_2 are denoted by delta1 and delta2 respectively.

Example 1 Consider the following fractional delay differential equation [15–23]

$$D^{\alpha} y(x) + y(x) + y(x-\tau) = \frac{2}{\Gamma(3-\alpha)} x^{2-\alpha} - \frac{1}{\Gamma(2-\alpha)} x^{1-\alpha} + 2\tau x - \tau^2 - \tau, \tag{34}$$

where, $x > 0, 0 < \alpha \leq 1$, with initial condition $y(0) = 0$ and the exact solution $y(x) = x^2 - x$ for $\alpha = 1$.

We have discussed this problem by taking $\tau(x) = 0.01e^{-x}$. In particular, taking $u = v = 0, u = v = \mp\frac{1}{2}, u = -v = \pm\frac{1}{2}$, we get Legendre polynomial, Chebyshev polynomial of first, second, third and fourth kind respectively.

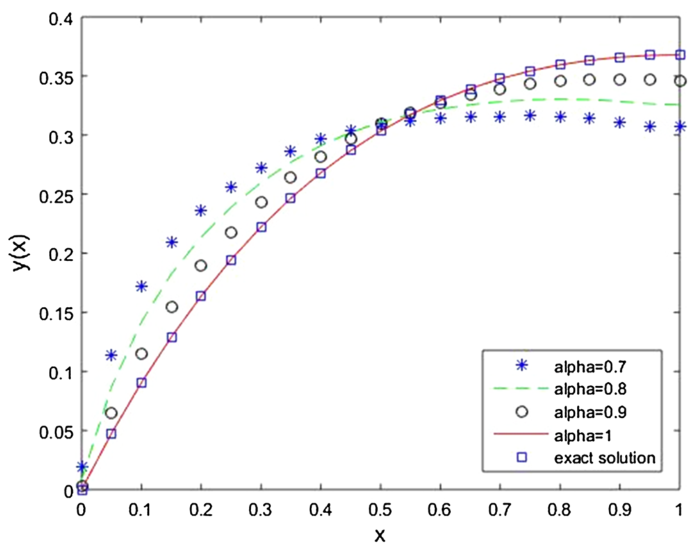

Fig. 9 The absolute errors with and without noises for u = 0, v = 0, Example 2

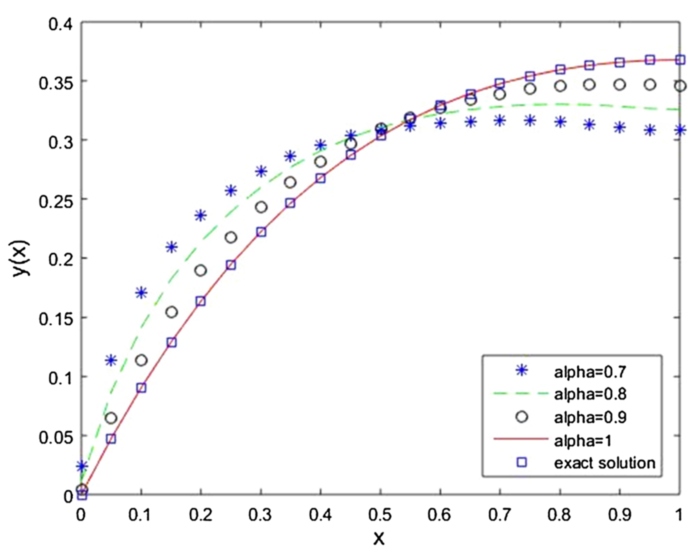

Fig. 10 The absolute errors with and without noises for u = 1/2, v = 1/2, Example 2

We have discussed Example 1 for different special cases and found numerical solution and exact solution are identical in each cases. We need only three basis elements to achieve exact solution by our proposed method.

Case 1: Taking $u = v = 0, n = 2$ then

$$C^T = [0, 1, 0], C^T J^{(1)} = \left[-\frac{1}{6}, 0, \frac{1}{6}\right]$$

and

$$y(x) = C^T J^{(1)} \mu^{(0,0)}(x) + A^T \mu^{(0,0)}(x) = x^2 - x,$$

is the exact solution.

Case 2: Taking $u = v = \frac{1}{2}, n = 2$ then

$$C^T = \left[0, \frac{2}{3}, 0\right], C^T J^{(1)} = \left[-\frac{3}{16}, 0, \frac{1}{10}\right]$$

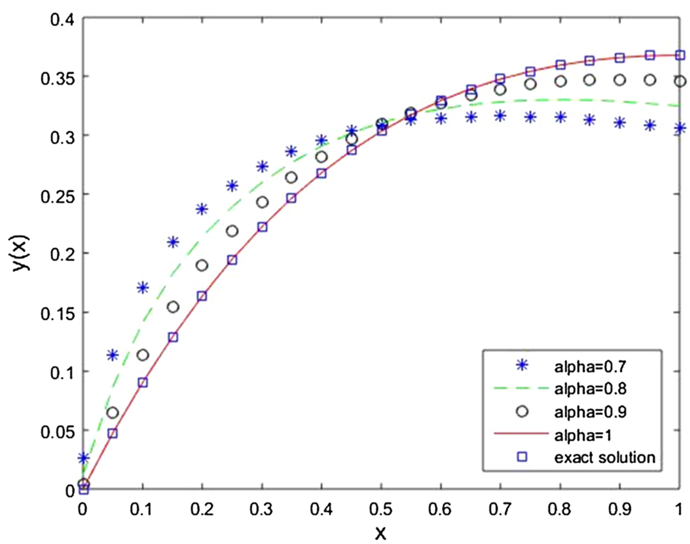

Fig. 11 The absolute errors with and without noises for u = $-1/2$, v = $1/2$, Example 2

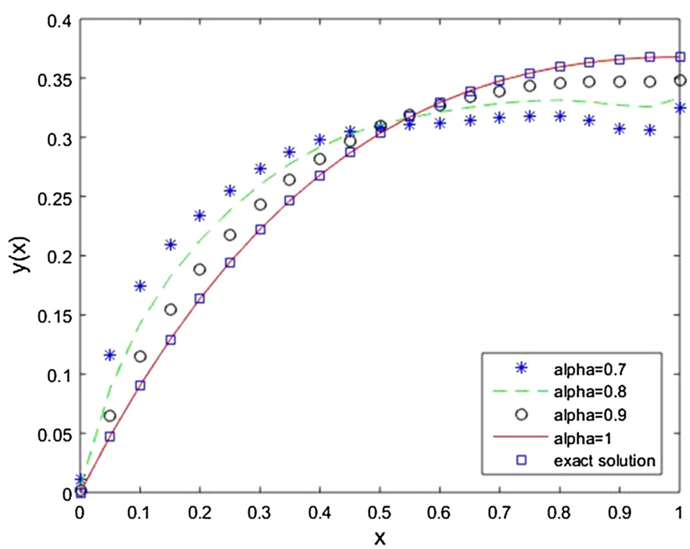

Fig. 12 The absolute errors with and without noises for u = $1/2$, v = $-1/2$, Example 2

Table 2 Comparisons of MAE and RMSE for different value of u and v at n = 5 and 7 for Example 2

u	v	Maximum absolute error		Root mean square error	
		n = 5	n = 7	n = 5	n = 7
0	0	4.5433×10^{-6}	3.3418×10^{-8}	4.5135×10^{-7}	3.1685×10^{-9}
$\frac{1}{2}$	$\frac{1}{2}$	4.6924×10^{-6}	4.1223×10^{-8}	4.8723×10^{-7}	3.9907×10^{-9}
$-\frac{1}{2}$	$\frac{1}{2}$	6.8701×10^{-6}	4.9867×10^{-8}	6.5811×10^{-7}	4.5774×10^{-9}
$\frac{1}{2}$	$-\frac{1}{2}$	3.1703×10^{-6}	2.1755×10^{-8}	3.1114×10^{-7}	2.1670×10^{-9}

Table 3 Comparison of results from our method and Chebyshev wavelet method in [18] for Example 2

x	Present method for n = 7			Method in [18]
	$u = 0,\ v = 0$	$u = 1/2, v = 1/2$	$u = -1/2, v = 1/2$	
0.01	4.7358×10^{-10}	1.2151×10^{-9}	2.2825×10^{-9}	8.2000×10^{-9}
0.02	1.1806×10^{-11}	4.9038×10^{-10}	1.0753×10^{-9}	6.6800×10^{-8}
0.03	3.0801×10^{-10}	2.5593×10^{-12}	2.2410×10^{-10}	2.2880×10^{-7}
0.04	4.6179×10^{-10}	3.1563×10^{-10}	3.4405×10^{-10}	5.5050×10^{-7}
0.05	5.1227×10^{-10}	4.9312×10^{-10}	6.9183×10^{-10}	1.0913×10^{-6}
0.06	4.9187×10^{-10}	5.7193×10^{-10}	8.7258×10^{-10}	1.9142×10^{-6}
0.07	4.2707×10^{-10}	5.8270×10^{-10}	9.3126×10^{-10}	3.0852×10^{-6}
0.08	3.3915×10^{-10}	5.5051×10^{-10}	9.0544×10^{-10}	4.6741×10^{-6}
0.09	2.4489×10^{-10}	4.9557×10^{-10}	8.2607×10^{-10}	6.7542×10^{-6}
0.10	1.5714×10^{-10}	4.3385×10^{-10}	7.1834×10^{-10}	9.4026×10^{-6}

Table 4 Comparison of results from our method and Laguerre wavelet method in [24] for Example 2

x	Present method for n =5			Method in [24] M =5
	$u = 0,\ v = 0$	$u = 1/2, v = 1/2$	$u = -1/2, v = 1/2$	
0.1	3.8435×10^{-7}	3.6604×10^{-7}	5.9004×10^{-7}	2.1700×10^{-8}
0.2	5.5264×10^{-8}	5.9731×10^{-8}	2.6351×10^{-7}	1.4823×10^{-6}
0.3	4.8734×10^{-8}	1.2034×10^{-7}	9.8401×10^{-8}	6.5389×10^{-6}
0.4	7.1498×10^{-7}	7.2015×10^{-7}	7.6043×10^{-7}	9.9500×10^{-6}
0.5	1.3449×10^{-6}	1.3479×10^{-6}	1.7396×10^{-6}	8.5111×10^{-6}
0.6	1.6621×10^{-6}	1.7520×10^{-6}	2.5392×10^{-6}	9.1161×10^{-5}
0.7	1.9793×10^{-6}	2.2126×10^{-6}	3.1933×10^{-6}	3.0880×10^{-4}
0.8	2.8579×10^{-6}	3.2043×10^{-6}	4.1752×10^{-6}	7.6648×10^{-4}
0.9	4.2169×10^{-6}	4.5101×10^{-6}	5.7570×10^{-6}	1.6087×10^{-3}
1	3.9441×10^{-6}	3.8363×10^{-6}	6.8701×10^{-6}	3.0244×10^{-3}

and

$$y(x) = C^T J^{(1)} \mu^{\left(\frac{1}{2},\frac{1}{2}\right)}(x) + A^T \mu^{\left(\frac{1}{2},\frac{1}{2}\right)}(x) = x^2 - x,$$

is the exact solution.

Case 3: Taking $u = -\frac{1}{2}, v = \frac{1}{2}, n = 2$ then

$$C^T = \left[\frac{1}{2}, 1, 0\right], C^T J^{(1)} = \left[-\frac{1}{8}, \frac{1}{8}, \frac{1}{6}\right]$$

and

$$y(x) = C^T J^{(1)} \mu^{\left(-\frac{1}{2},\frac{1}{2}\right)}(x) + A^T \mu^{\left(-\frac{1}{2},\frac{1}{2}\right)}(x) = x^2 - x,$$

is the exact solution.

Table 5 Comparison of results from our method ($u = 0$, $v = 0$) and Chebyshev wavelet method in [18] for Example 2 at different values of $\boldsymbol{\alpha} = \mathbf{2.25}$, 2.5 and 2.75

X	$\alpha = 2.25$		$\alpha = 2.5$		$\alpha = 2.75$	
	Present method	Method in [18]	Present method	Method in [18]	Present method	Method in [18]
0.00	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
0.01	0.99004	0.99005	0.99005	0.99001	0.99004	0.99004
0.02	0.98014	0.98020	0.98018	0.98020	0.98020	0.98020
0.03	0.97031	0.97046	0.97041	0.97045	0.97044	0.96079
0.04	0.96053	0.96083	0.96071	0.96080	0.96077	0.95123
0.05	0.95080	0.95131	0.95108	0.95125	0.95119	0.94176
0.06	0.94112	0.94191	0.94154	0.94180	0.94170	0.93239
0.07	0.93148	0.93263	0.93206	0.93245	0.93230	0.92311
0.08	0.92190	0.92346	0.92266	0.92321	0.92299	0.91392
0.09	0.91236	0.91442	0.91333	0.91406	0.91375	0.90482
0.10	0.90286	0.90552	0.90407	0.90501	0.90461	1.00000

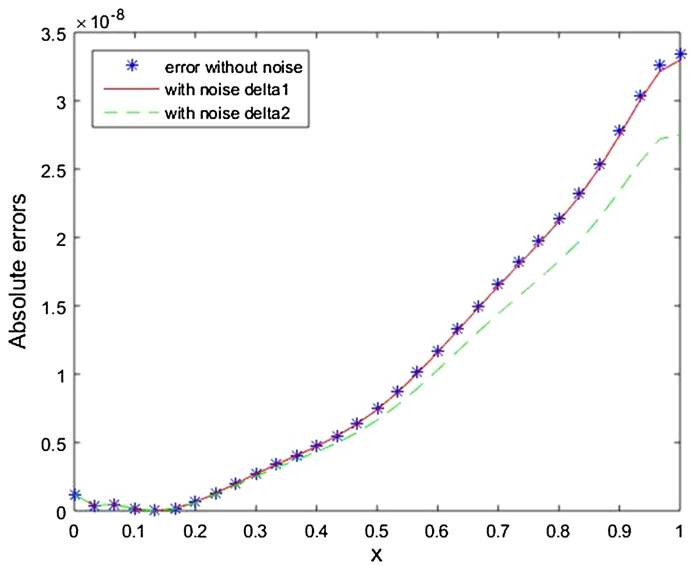

Fig. 13 The behaviour of solution for different values of α at u = 0, v = 0, Example 3

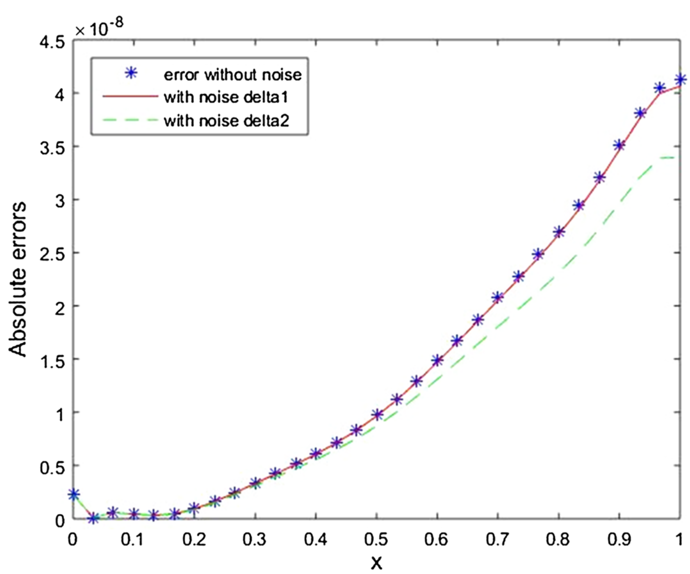

Fig. 14 The behaviour of solution for different values of α at u = 1/2, v = 1/2, Example 3

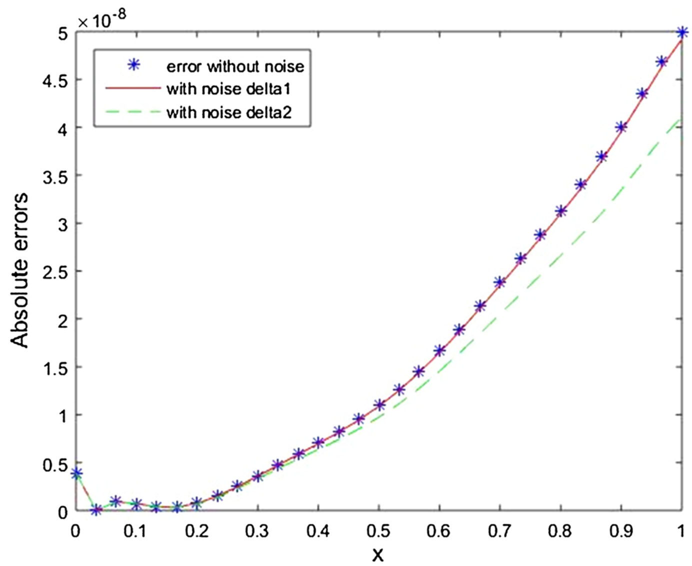

Fig. 15 The behaviour of solution for different values of α at u $= -1/2$, v $= 1/2$, Example 3

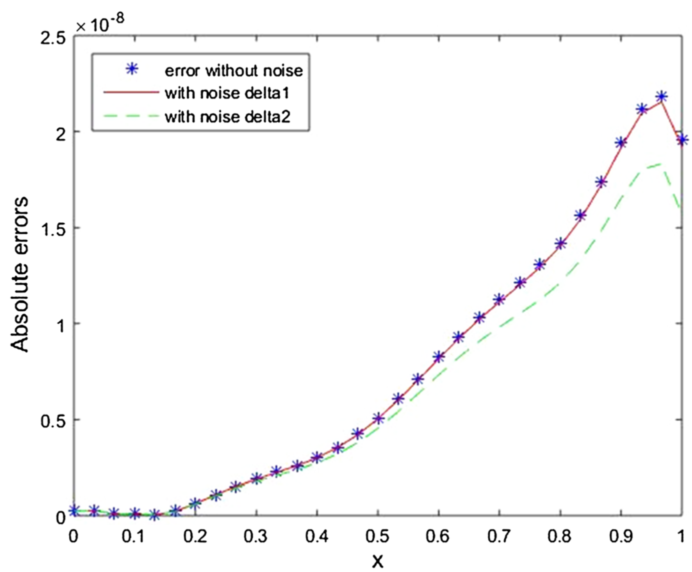

Fig. 16 The behaviour of solution for different values of α at u $= 1/2$, v $= -1/2$, Example 3

Case 4: Taking $u = \frac{1}{2}, v = -\frac{1}{2}, n = 2$ then

$$C^T = \left[-\frac{1}{2}, 1, 0\right], C^T J^{(1)} = \left[-\frac{1}{8}, -\frac{1}{8}, \frac{1}{6}\right]$$

and

$$y(x) = C^T J^{(1)} \mu^{\left(\frac{1}{2}, -\frac{1}{2}\right)}(x) + A^T \mu^{\left(\frac{1}{2}, -\frac{1}{2}\right)}(x) = x^2 - x,$$

is the exact solution.

From cases 1–4, it is observed that proposed method provides exact solution for the DDE given by Example 1. Whereas using existing methods [15–23], it is difficult to get exact solution. Therefore the proposed method is more accurate than the methods discussed in [15–23].

From Figs. 1, 2, 3 and 4, it is shown how the solution varies continuously for different values of fractional order derivative α for different particular cases of Jacobi polynomials respectively.

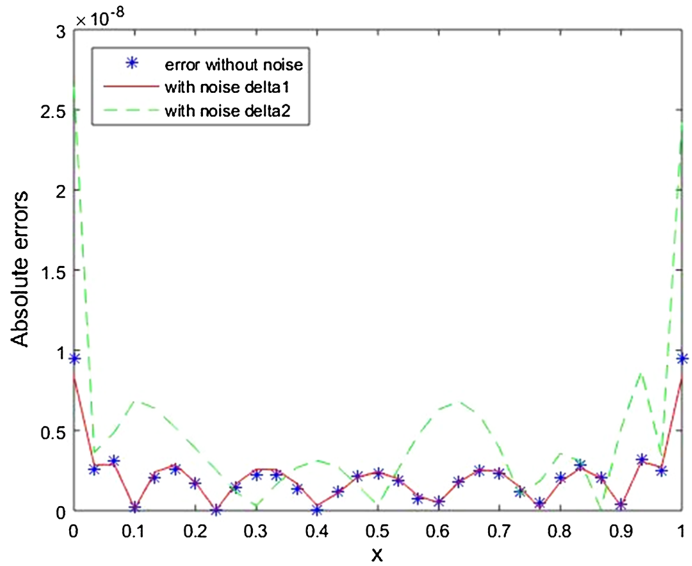

Fig. 17 The absolute errors with and without noises for u = 0, v = 0, Example 3

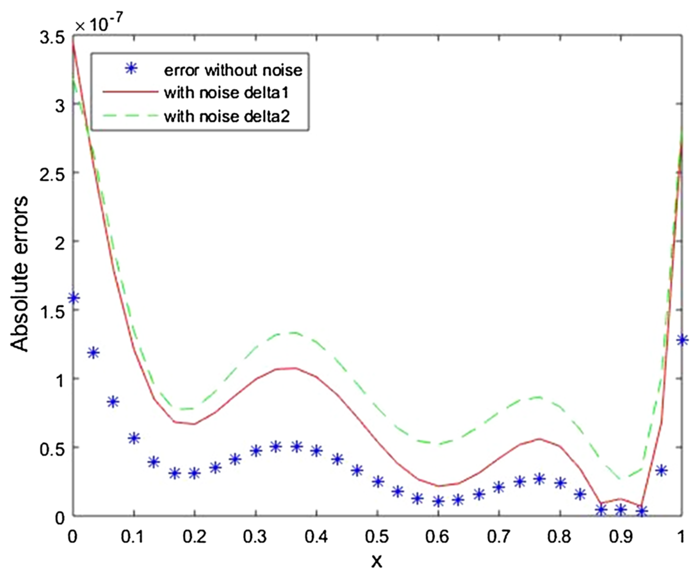

Fig. 18 The absolute errors with and without noises for u = 1/2, v = 1/2, Example 3

Example 2 Consider the FDDE [18,24]

$$D^{\alpha} y(x) + y(x) + y(x - 0.3) = e^{-x+0.3}, 0 \leq x \leq 1, 2 < \alpha \leq 3, \tag{35}$$

Subject to the initial condition $y(0) = 1$, $y'(0) = -1$, $y''(0) = 1$, and having exact solution as $y(x) = e^{-x}$ at $\alpha = 3$.

From Figs. 5, 6, 7 and 8, it is shown that solution varies continuously for Legendre polynomial, Chebyshev polynomial second kind, Chebyshev polynomial third kind, Chebyshev polynomial fourth kind respectively with different values of fractional order. For integer order solution coincide with the exact solution in all the cases.

We have taken to different random noises for stability of the method. In Figs. 9, 10, 11 and 12, we have plotted MAE with and without noises for Legendre polynomial, Chebyshev polynomial second kind, Chebyshev polynomial third kind, Chebyshev polynomial fourth kind respectively and it is observed that our approximating method is stable due to small changes in MAE.

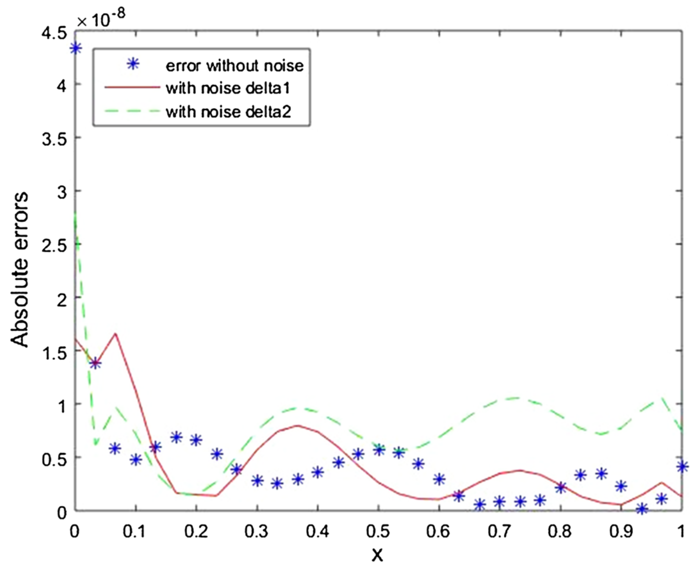

Fig. 19 The absolute errors with and without noises for u = −1/2, v = 1/2, Example 3

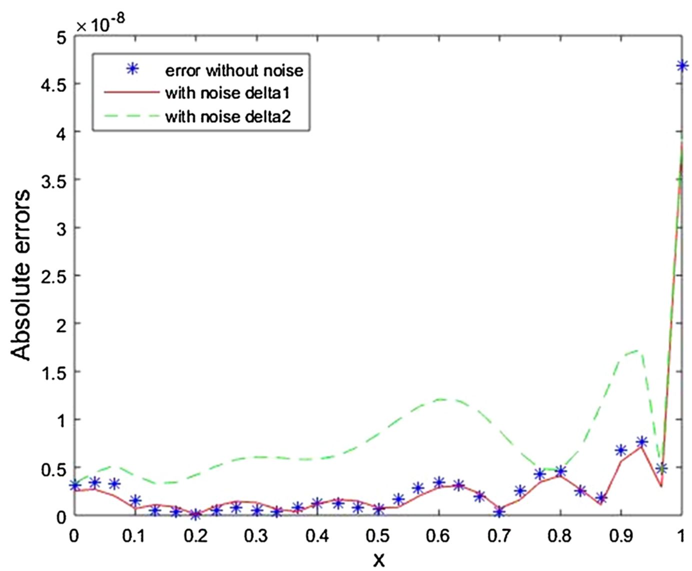

Fig. 20 The absolute errors with and without noises for u = 1/2, v = −1/2, Example 3

In Table 2, we have compared results for different polynomials and it is detected that results from Legendre polynomial and Chebyshev polynomial of fourth kind are quite better than the Chebyshev polynomial of second and third kind. The comparison of absolute errors from proposed method with the Chebyshev wavelets method [18] and Laguerre wavelets method [24] are shown in Tables 3 and 4 respectively.

From Tables 3 and 4, it is observed that our proposed method is more accurate.

In Table 5, we have listed the approximate solutions by our proposed method and Chebyshev wavelets method [18] for different values of α.

From Table 5, it is observed that our method shows good agreement with method in [18] for different fractional order.

Example 3 Consider the FDDE [15,16],

$$D^{\alpha}y(x) = -y(x) + 0.1y(0.8x) + 0.5D^{\alpha}y(0.8x) + (0.32x - 0.5)e^{0.8x} + e^{-x},$$
$$x \geq 0, 0 < \alpha \leq 1, \tag{36}$$

Subject to the initial condition $y(0) = 0$ and having exact solution as $y(x) = xe^{-x}$ at $\alpha = 1$.

Table 6 Comparisons of MAE and RMSE for different value of u and v at n = 5 and 7 for Example 3

u	v	Maximum absolute error		Root mean square error	
		n = 5	n = 7	n = 5	n = 7
0	0	4.9897×10^{-6}	9.5177×10^{-9}	3.7229×10^{-7}	6.4802×10^{-10}
$\frac{1}{2}$	$\frac{1}{2}$	7.8676×10^{-6}	1.5915×10^{-7}	4.5137×10^{-7}	1.1596×10^{-8}
$-\frac{1}{2}$	$\frac{1}{2}$	1.3898×10^{-5}	4.3297×10^{-8}	7.6121×10^{-7}	2.3553×10^{-9}
$\frac{1}{2}$	$-\frac{1}{2}$	1.5160×10^{-5}	4.6848×10^{-8}	3.2932×10^{-7}	6.1652×10^{-10}

Table 7 Comparison of results from our method and methods in [15,16] for Example 3

x	Present method for n = 7		Method in [15]	Method in [16]
	$\boldsymbol{u = 0, v = 0}$	$\boldsymbol{u = 1/2, v = 1/2}$		
0.1	2.60 E−10	5.63 E−08	1.42 E−4	8.68 E−4
0.2	1.67 E−09	3.13 E−08	1.17 E−4	1.49 E−3
0.3	2.27 E−09	4.71 E−08	9.45 E−4	1.90 E−3
0.4	7.86 E−11	4.73 E−08	7.59 E−4	2.16 E−3
0.5	2.36 E−09	2.52 E−08	6.03 E−4	2.28 E−3
0.6	6.12 E−10	1.09 E−08	4.73 E−4	2.31 E−3
0.7	2.28 E−09	2.07 E−08	3.64 E−4	2.27 E−3
0.8	2.06 E−09	2.38 E−08	2.75 E−4	2.17 E−3
0.9	3.66 E−10	4.95 E−09	2.03 E−4	2.03 E−3
1.0	9.52 E−09	1.28 E−07	1.43 E−4	1.86 E−3

From Figs. 13, 14, 15 and 16, it is shown that solution varies continuously for different cases of Jacobi polynomial with different values of fractional order and for $\alpha = 1$, the approximate solution coincides with the exact solution in all the cases.

For Example 3, as similar in two we have taken to different random noises for stability of the method. In Figs. 17, 18, 19 and 20, we have plotted MAE with and without noises for all four kind of polynomials and it is observed that our approximating method is stable due to small changes in MAE.

In Table 6, we have compared results for different polynomials and it is noticed that results from Legendre polynomial are better than the Chebyshev polynomials. The comparison of absolute errors from proposed method with the Reproducing kernel Hilbert space method (RKHSM) [15] and Runge–Kutta type method (RKTM) [16] are provided in Table 7.

From Table 7, it is observed that obtained numerical results using proposed are comparatively better than RKHSM and RKTM.

8 Conclusions

It is shown that results obtained from proposed method are attractive and comparatively better to the existing numerical methods [15–23] discussed in literature. The presented method has also been found stable even through high noise level. We observed that the obtained approximate solution varies continuously from fractional order DDEs to integer order DDEs. The results obtained using proposed method are found more accurate in comparison to results obtained from other methods (shown in Tables 3, 4 and 7). Further, it is also observed that the proposed method works well and produces comparatively better results using only few number of bases elements approximating the unknown function. For each test problem four polynomials are used and obtained results are compared in Tables 2 and 6. We observe that the choice of Legendre polynomial and Chebyshev polynomial of fourth kind approximating the unknown function performs better than the Chebyshev second and third kind polynomials.

References

1. H. Smith, *An Introduction to Delay Differential Equations with Applications to the Life Sciences* (Springer, New York, 2011)
2. V. Lakshmikantham, S. Leela, *Differential and Integral Inequalities* (Academic Press, New York, 1969)
3. W.G. Aiello, H.I. Freedman, A time-delay model of single-species growth with stage structure. Math. Biosci. **101**, 139 (1990)
4. A.R. Davis, A. Karageorghis, T.N. Phillips, Spectral Galerkin methods for the primary two-point bour problem in modelling viscoelastic flows. Int. J. Numer. Methods Eng. **26**, 647 (1988)
5. S.A. Gourley, Y. Kuang, A stage structured predator-prey model and its dependence on maturation delay and death rate. J. Math. Biol. **49**, 188 (2004)
6. J. Li, Y. Kuang, C. Mason, Modelling the glucose-insulin regulatory system and ultradian insulin secretory oscillations with two time delays. J. Theor. Biol. **242**, 722 (2006)
7. A.D. Robinson, The use of control systems analysis in neurophysiology of eye movements. Ann. Rev. Neurosci. **4**, 462 (1981)
8. R.L. Bagley, P.J. Torvik, A theoretical basis for the application of fractional calculus to viscoelasticity. J. Rheol. **27**, 201 (1983)
9. R.L. Bagley, P.J. Torvik, Fractional calculus a differential approach to the analysis of viscoelasticity damped structures. AIAA J. **21**, 741 (1983)
10. R.L. Bagley, P.J. Torvik, Fractional calculus in the transient analysis of viscoelasticity damped structures. AIAA J. **23**, 918 (1985)
11. R. Panda, M. Dash, Fractional generalized splines and signal processing. Signal Process **86**, 2340 (2006)
12. R.L. Magin, Fractional calculus in bioengineering. Crit. Rev. Biomed. Eng. **32**, 1 (2004)
13. G.W. Bohannan, Analog fractional order controller in temperature and motor control applications. J. Vib. Control **14**, 1487 (2008)
14. C.E. Falbo, Some Elementary Methods for Solving Functional Differential Equations. http://www.mathfile.net/hicstatFDE.pdf
15. X. Lv, Y. Gao, The RKHSM for solving neutral functional differential equations with proportional delays. Math. Methods Appl. Sci. **36**, 642 (2013)
16. W. Wang, Y. Zhang, S. Li, Stability of continuous Runge–Kutta type methods for nonlinear neutral delay-differential equations. Appl. Math. Model. Simul. Comput. Eng. Environ. Syst. **33**(8), 3319 (2009)
17. U. Saeed, M. ur Rehman, Hermite wavelet method for fractional delay differential equations. J. Differ. Equ. Article ID **359093**, 8 (2014)
18. M.A. Iqbal, A. Ali, S.T. Mohyud-Din, Chebyshev wavelets method for fractional delay differential equations. Int. J. Mod. Appl. Phys. **4**(1), 49 (2013)
19. D.J. Evans, K.R. Raslan, The Adomian decomposition method for solving delay differential equation. Int. J. Comput. Math. **82**(1), 49 (2005)
20. B.P. Mohaddam, Z.S. Mostaghim, A numerical method based on finite difference for solving fractional delay differential equations. J. Taibah Univ. Sci. **7**, 120 (2013)
21. Z. Wang, A numerical method for delayed fractional-order differential equations. J. Appl. Math. Article ID **256071**, 7 (2013)
22. Z. Wang, X. Huang, J. Zhou, A numerical method for delayed fractional-order differential equations: based on GL definition. Appl. Math. **7**(2), 525 (2013)
23. R.K. Pandey, N. Kumar, N. Mohaptra, An approximate method for solving fractional delay differential equations. Int. J. Appl. Comput. Math. (2016). doi:10.1007/s40819-016-0186-3
24. M.A. Iqbal, U. Saeed, S.T. Mohyud-Din, Modified Laguerre wavelets method for delay differential equation of fractional-order. Egypt. J. Basic Appl. Sci. **2**, 50 (2015)
25. E.H. Doha, A.H. Bhrawy, D. Baleanu, S.S. Ezz-Eldien, The operational matrix formulation of the Jacobi tau approximation for space fractional diffusion equation. Adv. Differ. Equ. (2014). http://www.advancesindifferenceequations.com/content/2014/1/231
26. A. Ahmadian, M. Suleiman, S. Salahshour, D. Baleanu, A Jacobi operational matrix for solving a fuzzy linear fractional differential equation. Adv. Differ. Equ. (2014). http://www.advancesindifferenceequations.com/content/2013/1/104
27. A.H. Bhrawy, M.M. Tharwat, M.A. Alghamdi, A new operational matrix of fractional integration for shifted Jacobi polynomials. Bull. Malays. Math. Sci. Soc. (2) **37**(4), 983 (2014)
28. S. Kazem, An integral operational matrix based on Jacobi polynomials for solving fractional-order differential equations. Appl. Math. Model. **37**, 1126 (2013)
29. R.K. Pandey, S. Suman, K.K. Singh, O.P. Singh, Approximate solution of Abel inversion using Chebshev polynomials. Appl. Math. Comput. **237**, 120 (2014)