Hindawi
Advances in Mathematical Physics
Volume 2020, Article ID 8101843, 12 pages
https://doi.org/10.1155/2020/8101843

Research Article

Using Reproducing Kernel for Solving a Class of Fractional Order Integral Differential Equations

Zhiyuan Li, Meichun Wang, Yulan Wang, and Jing Pang

Department of Mathematics, Inner Mongolia University of Technology, Hohhot 010051, China

Correspondence should be addressed to Yulan Wang; wylnei@163.com

Received 30 December 2019; Accepted 24 February 2020; Published 11 March 2020

Academic Editor: Zengtao Chen

This paper is devoted to the numerical scheme for a class of fractional order integrodifferential equations by reproducing kernel interpolation collocation method with reproducing kernel function in the form of Jacobi polynomials. Reproducing kernel function in the form of Jacobi polynomials is established for the first time. It is implemented as a reproducing kernel method. The numerical solutions obtained by taking the different values of parameter are compared; Schmidt orthogonalization process is avoided. It is proved that this method is feasible and accurate through some numerical examples.

1. Introduction

In this paper, the reproducing kernel interpolation collocation method with reproducing kernel function in the form of Jacobi polynomials is applied to solve the following linear fractional integrodifferential equations (FIDEs):

$$\begin{cases} D^{\mu}u_1(x) + \int_0^t k_{11}(x,t)u_1(t) + k_{12}(x,t)u_2(t)dt = f_1(x), \\ D^{\mu}u_2(x) + \int_0^t k_{21}(x,t)u_1(t) + k_{22}(x,t)u_2(t)dt = f_2(x), \\ 0 < x, t \le 1, u_1(0) = 0, u_2(0) = 0, \end{cases} \tag{1}$$

where $0 < \mu \le 1$, $f_n(x), n = 1, 2,$ and $k_{ij}(x,t)$, $i, j = 1, 2,$ are given functions. $D^{\mu}u_n(x)$ indicates that μ is the Caputo fractional derivative defined by $u_n(x)$, $n = 1, 2$.

Fractional order integrodifferential equation appears in the formulation process of applied science, such as physics and finance. However, it is very difficult to obtain the analytic solution of linear integrodifferential equations of fractional order, so many researchers try their best to study numerical solution of linear FIDEs and system of linear FIDEs in recent years [1–5]. Since the reproducing kernel method can not only obtain the exact solution in the form of series but also obtain the approximate solution with higher accuracy, the method has been widely used in linear and nonlinear problems, integral and differential equations, fractional partial differential equation, and so on [6–15]. But there are no scholars that use the reproducing kernel interpolation collocation method to solve the linear integrodifferential equations of fractional order. In this paper, linear integrodifferential equations of fractional order are solved by the reproducing kernel interpolation collocation method with reproducing kernel function in the form of Jacobi polynomials for the first time. The fractional derivative is described in the Caputo sense.

2. Preliminaries

Definition 1. The Caputo fractional derivative operator of order $0 < \mu \le 1$ is defined as

$$D^{\mu}u(t) = \begin{cases} \dfrac{1}{\Gamma(1-\mu)}\displaystyle\int_0^t (t-\tau)^{-\mu}\frac{\partial u(\tau)}{\partial \tau}d\tau, & 0 < \mu < 1, \\ \dfrac{\partial u(t)}{\partial t}, & \alpha = 1. \end{cases} \tag{2}$$

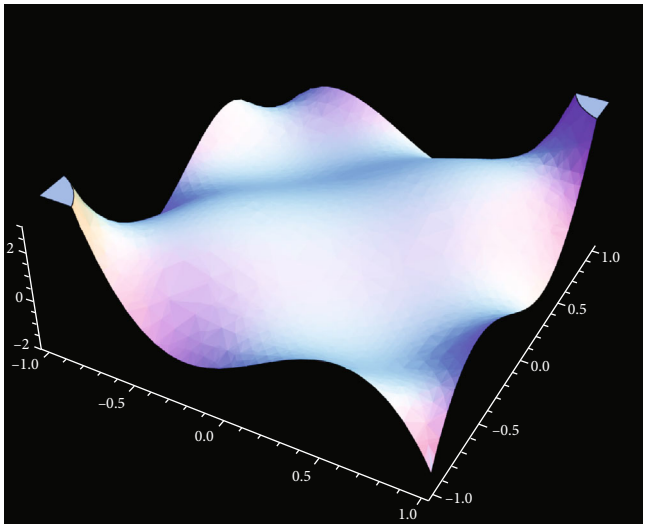

Figure 1: Reproducing kernel $R_x(y)$ with $n = 3, \alpha = \beta = 0$.

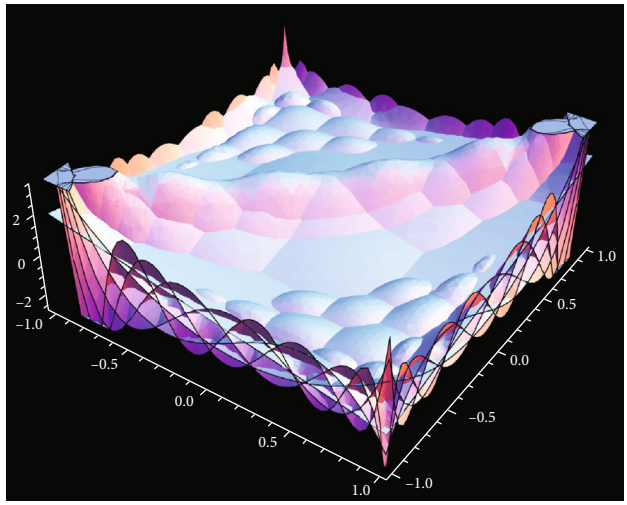

Figure 2: The set of reproducing kernel $R_x(y)$ with $n = 1, 2 \cdots, 7$, $\alpha = \beta = 0$.

Definition 2. Let H be the real Hilbert spaces of functions $f : \Omega \longrightarrow R$. A function $K : \Omega \times \Omega \longrightarrow R$ is called reproducing kernel for H if

(i) $K(x, \cdot) \in H$ for all $x \in \Omega$

(ii) $f(x) = \langle f, K(\cdot, x)\rangle_H$ for all $f \in H$ and all $x \in \Omega$

2.1. The Shifted Jacobi Polynomials. The shifted Jacobi polynomials $P_{1,i}^{\alpha,\beta}(x)$ of degree i is given [16] by

$$P_{1,i}^{\alpha,\beta}(x) = \sum_{k=0}^{i} (-1)^{(i-k)} \frac{\Gamma(i+\beta+1)\Gamma(i+k+1+\alpha+\beta)}{\Gamma(k+1+\beta)\Gamma(i+\alpha+\beta+1)(i-k)!k!} x^k, \tag{3}$$

where

$$P_{1,i}^{\alpha,\beta}(0) = (-1)^i \frac{\Gamma(i+\beta+1)}{\Gamma(1+\beta)i!}, \qquad P_{1,i}^{\alpha,\beta}(1) = \frac{\Gamma(i+\alpha+1)}{\Gamma(1+\alpha)i!}. \tag{4}$$

The shifted Jacobi polynomials on the interval $x \in [0, 1]$ are orthogonal with the orthogonality condition which is

$$\int_0^1 P_{1,n}^{\alpha,\beta}(x) P_{1,m}^{\alpha,\beta}(x) \omega^{\alpha,\beta}(x) dx = h_k, \tag{5}$$

where $\omega(x) = x^\beta (1-x)^\alpha$ is a weight function, and

$$h_k = \begin{cases} \dfrac{\Gamma(k+\alpha+1)\Gamma(k+\beta+1)}{(2k+\alpha+\beta+1)k!\Gamma(k+\alpha+\beta+1)}, & n = m, \\ 0, & n \neq m. \end{cases} \tag{6}$$

2.2. Reproducing Kernel Space

Definition 3. Let

$$H_n[0,1] = \left\{ P_{1,i}^{\alpha,\beta}(x) \,\middle|\, \int_0^1 \omega(x) \left| P_{1,n}^{\alpha,\beta}(x) \right|^2 dx < \infty, i = 1, \cdots, n \right\} \tag{7}$$

be the weighted inner product space of the shifted Jacobi polynomials on $[0, 1]$. The inner product and norm are defined as

$$\left\langle P_{1,m}^{\alpha,\beta}(x), P_{1,n}^{\alpha,\beta}(x) \right\rangle = \int_0^1 \omega(x) P_{1,m}^{\alpha,\beta}(x) P_{1,n}^{\alpha,\beta}(x) dx, \qquad \left\| P_{1,n}^{\alpha,\beta}(x) \right\|_{H_n} = \sqrt{\left\langle P_{1,n}^{\alpha,\beta}(x), P_{1,n}^{\alpha,\beta}(x) \right\rangle_{H_n}}, \tag{8}$$

$$\forall P_{1,n}^{\alpha,\beta}(x), P_{1,m}^{\alpha,\beta}(x) \in H_n[0,1].$$

Let $L^2[0,1] = \{f(x) | \int_0^1 \omega(x) |f(x)|^2 dx < \infty\}$. From [17–20], we can prove that $H_n[0, 1]$ is a reproducing kernel Hilbert space. Its reproducing kernel is

$$R(x, y) = R_x(y) = \sum_{i=0}^{n} e_i(x) e_i(y), \tag{9}$$

where $e_i(x) = \sqrt{((2k+\alpha+\beta+1)k!\Gamma(k+\alpha+\beta+1))/(\Gamma(k+\alpha+1)\Gamma(k+\beta+1))} P_{1,i}^{\alpha,\beta}(x)$. Reproducing kernel is shown in Figures 1–4.

Definition 4. Let

$$\bar{H}_n[0,1] = \{u \mid u \in H_n[0,1], \quad u(0) = 0\}. \tag{10}$$

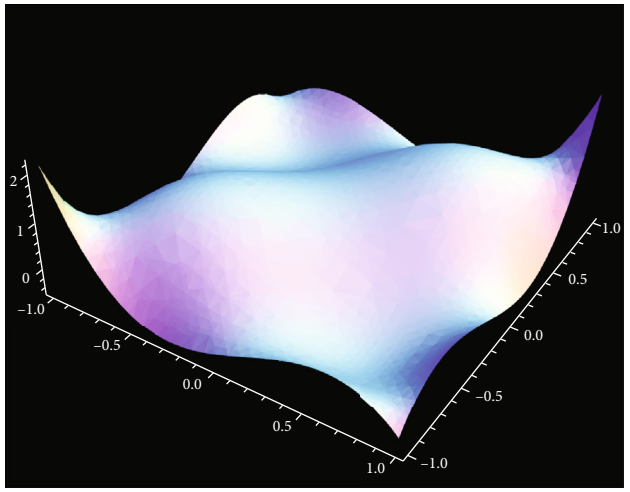

FIGURE 3: Reproducing kernel $R_x(y)$ with $n = 3, \alpha = \beta = -0.5$.

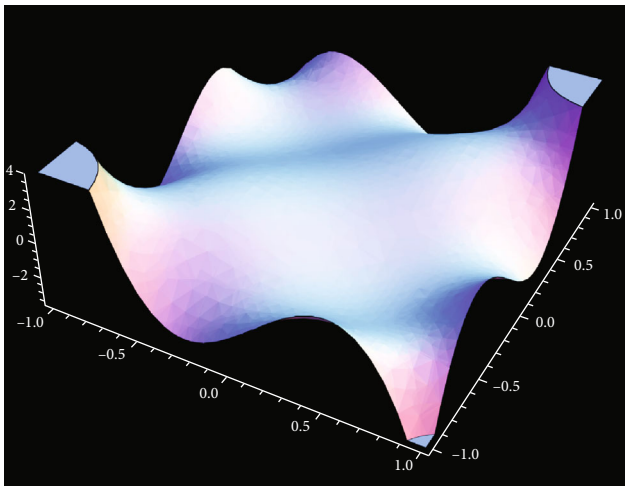

FIGURE 4: Reproducing kernel $R_x(y)$ with $n = 3, \alpha = \beta = 0.5$.

Its norm is the same as the norm of $H_n[0,1]$. It can easily be shown that $\bar{H}_n[0,1]$ is a reproducing kernel Hilbert space. According to [18–22], the reproducing kernel of $\bar{H}_n[0,1]$ is

$$K(x,y) = K_x(y) = R(x,y) - \frac{R(0,x)R(y,0)}{\|R(0,0)\|^2}. \tag{11}$$

Definition 5. The inner product space is defined as

$$\bar{H}_n[0,1] \oplus \bar{H}_n[0,1] = \left\{ U(x) = [u_1(x), u_2(x)]^T \,\middle|\, u_1(x), u_2(x) \in \bar{H}_n[0,1] \right\}. \tag{12}$$

Its inner product and norm are defined by

$$\begin{aligned} \langle U(x), V(x) \rangle &= \sum_{i=1}^{2} \langle u_i(x), v_i(x) \rangle_{\bar{H}_n[0,1] \oplus \bar{H}_n[0,1]}, \\ \|U(x)\|^2 &= \sum_{i=1}^{2} \|u_i(x)\|_{\bar{H}_n[0,1] \oplus \bar{H}_n[0,1]}. \end{aligned} \tag{13}$$

It is easy to verify that $\bar{H}_n[0,1] \oplus \bar{H}_n[0,1]$ is a Hilbert space with the definition of inner product (13). Similarly, $L^2[0,1] \oplus L^2[0,1]$ is also a Hilbert space.

3. The Reproducing Kernel Interpolation Collocation Method

To solve equation (1), let

$$\begin{cases} l_{11}u_1 = D^{\mu}u_1(x) + \int_0^t k_{11}(x,t)u_1(t)dt, \\ l_{12}u_2 = \int_0^t k_{12}(x,t)u_2(x,t)dt, \\ l_{21}u_1 = \int_0^t k_{21}(x,t)u_1(t)dt, \\ l_{22}u_2 = D^{\mu}u_2(x) + \int_0^t k_{22}(x,t)u_2(t)dt. \end{cases} \tag{14}$$

So, equation (1) can be turned into

$$LU(x) = F(x), \tag{15}$$

where

$$L = \begin{bmatrix} l_{11} & l_{12} \\ l_{21} & l_{22} \end{bmatrix}. \tag{16}$$

The operator $L : \bar{H}_n[0,1] \oplus \bar{H}_n[0,1] \longrightarrow L^2[0,1] \oplus L^2[0,1]$ is a bounded linear operator.

Assuming that $\{x_i\}_{i=1}^{\infty}$ is dense on the interval $[0,1]$, put $\phi_{ijk} = l_{ij}^* k_{x_k}(x)$, where l_{ij}^* is the adjoint operator of l_{ij}. From [23–25], we have

$$\phi_{ijk}(x) = l_{ij}K_x(x_k), \quad i,j = 1,2, k = 1,2,\cdots. \tag{17}$$

Putting

$$\begin{aligned} \Psi_{i1}(x) &= (\phi_{11i}(x), \phi_{12i}(x))^T, \\ \Psi_{i2}(x) &= (\phi_{21i}(x), \phi_{22i}(x))^T, \\ & i, = 1,2,\cdots. \end{aligned} \tag{18}$$

Theorem 6. *For each fixed n, $\{\Psi_{ij}\}_{(1,1)}^{(n,2)}$ is linearly independent in $\bar{H}_n[0,1] \oplus \bar{H}_n[0,1]$.*

Proof. Letting

$$0 = \sum_{i=1}^{\infty}(c_{i1}\Psi_{i1}(x) + c_{i2}\Psi_{i2}(x)),$$
$$U_k(x) = [u_{k,1}(x), u_{k,2}(x)]^T, \tag{19}$$

where $u_{k,1}(x) \in L^2[0,1]$, when $x = x_1, x_2, \cdots, x_{k-1}, x_{k+1}, \cdots x_n$, $u_{k,1}(x) = 0$. But $u_{k,1}(x) \neq 0$, when x take other value, $0 \doteq u_{k,2}(x) \in L^2[0,1]$.

When $U_k \in \bar{H}_n[0,1] \oplus \bar{H}_n[0,1]$, we have $Lu_k = F_k$. So,

$$\begin{aligned}
0 &= \left\langle U_k, \sum_{i=1}^{n}(c_{i1}\Psi_{i1}(x) + c_{i2}\Psi_{i2}(x)) \right\rangle \\
&= \sum_{i=1}^{n}(c_{i1}\langle U_k, \Psi_{i1}\rangle + c_{i2}\langle U_k, \Psi_{i2}\rangle) \\
&= \sum_{i=1}^{n}\left(c_{i1}\left(\langle u_{k,1}, \phi_{11i}\rangle + \langle u_{k,2}, \phi_{12i}\rangle\right)\right. \\
&\quad \left. + c_{i2}\left(\langle u_{k,1}, \phi_{21i}\rangle + \langle u_{k,2}, \phi_{22i}\rangle\right)\right) \\
&= \sum_{i=1}^{n}(c_{i1}(l_{11}u_{k,1}(x_i) + l_{12}u_{k,2}(x_i)) \\
&\quad + c_{i2}(l_{21}u_{k,1}(x_i) + l_{22}u_{k,2}(x_i))) \\
&= \sum_{i=1}^{n}(c_{i1}u_{k,1}(x_i) + c_{i2}l_{21}u_{k,1}(x_i)) = c_{k1}u_{k,1}(x_k).
\end{aligned} \tag{20}$$

So, $c_{k1} = 0, k = 1, 2, \cdots, n$. Similarly, we have $c_{k2} = 0$.

Theorem 7. *$\{\Psi_{ij}\}_{(1,1)}^{(\infty,2)}$ is complete in space in $\bar{H}_n[0,1] \oplus \bar{H}_n[0,1]$.*

Proof. For each

$$U(x) = [u_1(x), u_2(x)]^T \in \bar{H}_n[0,1] \oplus \bar{H}_n[0,1], \tag{21}$$

it follows that $\langle U(x), \Psi_{ij}(x)\rangle = 0$, for every $i = 1, 2, \cdots, j = 1, 2$.

$$\begin{aligned}
0 &= \langle U(x), \Psi_{i1}(x)\rangle_{\bar{H}_n[0,1]\oplus\bar{H}_n[0,1]} \\
&= \left\langle u_1(x), l_{11}^* K_{x_i}(x)\right\rangle_{\bar{H}_n[0,1]} + \left\langle u_2(x), l_{12}^* K_{x_i}(x)\right\rangle_{\bar{H}_n[0,1]} \\
&= l_{11}u_1(x_i) + l_{12}u_2(x_i), \\
0 &= \langle U(x), \Psi_{i2}(x)\rangle_{\bar{H}_{(n,n)}[0,1]=\left\langle u_1(x), l_{21}^* K_{x_i}(x)\right\rangle_{\bar{H}_n[0,1]} + \left\langle u_2(x), l_{22}^* K_{x_i}(x)\right\rangle_{\bar{H}_n[0,1]}} \\
&= l_{21}u_1(x_i) + l_{22}u_2(x_i).
\end{aligned} \tag{22}$$

Since equation (1) has a unique solution, it follows that $U(X) = 0$.

The exact solution of equation (1) can be expressed as

$$U(X) = \sum_{i=1}^{\infty}\sum_{j=1}^{2} c_{ij}\Psi_{ij}(x), \tag{23}$$

and truncating the infinite series of the analytic solution, we obtain the approximate solution of equation (1).

$$U_m(X) = \sum_{i=1}^{m}\sum_{j=1}^{2} c_{ij}\Psi_{ij}(x). \tag{24}$$

Theorem 8. *Let $U \in \bar{H}_n[0,1] \oplus \bar{H}_n[0,1]$ be the exact solution of equation (1), U_m be the approximate solution of U, then U_m converges uniformly to U.*

Proof.

$$\begin{aligned}
\left|u_1(x) - u_{1,m}(x)\right| &= \left|\left\langle u_1 - u_{1,m}, K_x\right\rangle\right| \\
&\leq \left\|u_1 - u_{1,m}\right\|_{\bar{H}_n[0,1]}\|K_x\|_{\bar{H}_n[0,1]} \\
&\leq M\left\|u_1 - u_{1,m}\right\|_{\bar{H}_n[0,1]}.
\end{aligned} \tag{25}$$

Similarly,

$$\left|u_2(x) - u_{2,m}(x)\right| \leq M\left\|u_2 - u_{2,m}\right\|_{\bar{H}_n[0,1]}. \tag{26}$$

If we can obtain the coefficients of each $\Psi_{ij}(x)$, the approximate solution $U_m(x)$ can be obtained as well. Using $\Psi_{ij}(x)$ to do the inner products with both sides of equation (24), we have

$$\begin{cases}
\sum_{i=1}^{m} c_{i1}\langle\Psi_{i1}, \Psi_{n1}\rangle + \sum_{j=1}^{m} c_{j2}\left\langle\Psi_{j2}, \Psi_{n1}\right\rangle = f_1(x_n), \quad n = 1, 2, \cdots, m, \\
\sum_{i=1}^{m} c_{i1}\langle\Psi_{i1}, \Psi_{n2}\rangle + \sum_{j=1}^{m} c_{j2}\left\langle\Psi_{j2}, \Psi_{n2}\right\rangle = f_2(x_n), \quad n = 1, 2, \cdots, m.
\end{cases} \tag{27}$$

Letting

$$L_{2m} = \begin{bmatrix}
\langle\Psi_{i1}, \Psi_{m1}\rangle & \cdots & \left\langle\Psi_{j2}, \Psi_{m1}\right\rangle \\
\cdots & \cdots & \cdots \\
\langle\Psi_{i1}, \Psi_{m2}\rangle & \cdots & \left\langle\Psi_{j2}, \Psi_{m2}\right\rangle
\end{bmatrix}_{i,j,n=1,2,\cdots,m,}$$
$$F = (f_1(x_1), \cdots, f_1(x_m), f_2(x_1), \cdots, f_1(x_m))^T. \tag{28}$$

It is obvious that the inverse of A_{2m} exists by Theorem 6. So, we have

$$(c_{11}, c_{12}, \cdots, c_{1m}, c_{21}, c_{22}, \cdots, c_{2m})^T = L_{2m}^{-1}F. \tag{29}$$

Table 1: The numerical solutions of $u_1(x)$ for Example 1.

x	Exact solution	Approximate solution	$\lvert u_1(x) - u_{1,10}(x)\rvert$	$\left\lvert \frac{u_1(x) - u_{1,10}(x)}{u_1(x)} \right\rvert$
0.1	0.099	0.099	$6.85563E-15$	$7.37345E-14$
0.2	0.192	0.192	$1.05471E-15$	$3.15141E-14$
0.3	0.273	0.273	$4.77396E-15$	$1.22003E-15$
0.4	0.336	0.336	$3.05311E-15$	$3.48597E-14$
0.5	0.375	0.375	$1.24345E-14$	$7.10543E-15$
0.6	0.384	0.384	$1.09912E-14$	$8.67362E-16$
0.7	0.357	0.357	$5.43454E-14$	$1.14910E-13$
0.8	0.288	0.288	$1.09912E-14$	$7.17019E-14$
0.9	0.171	0.171	$1.87628E-14$	$2.66194E-14$

Table 2: The numerical solutions of $u_2(x)$ for Example 1.

x	Exact solution	Approximate solution	$\lvert u_2(x) - u_{2,10}(x)\rvert$	$\left\lvert \frac{u_2(x) - u_{2,10}(x)}{u_2(x)} \right\rvert$
0.1	−0.09	−0.09	$3.55271E-15$	$9.15934E-14$
0.2	−0.16	−0.16	$3.85803E-15$	$7.11237E-15$
0.3	−0.21	−0.21	$6.82787E-15$	$3.46284E-14$
0.4	−0.24	−0.24	$5.55112E-16$	$2.33609E-14$
0.5	−0.25	−0.25	$8.21565E-15$	$1.17684E-14$
0.6	−0.24	−0.24	$8.82627E-15$	$9.25186E-16$
0.7	−0.21	−0.21	$1.88183E-14$	$4.83740E-14$
0.8	−0.16	−0.16	$3.66374E-15$	$2.35922E-14$
0.9	−0.09	−0.09	$9.88098E-15$	$2.71388E-14$

4. Numerical Experiment

Example 1. We consider the following linear integrodifferential equations of fractional order [5]:

$$\begin{cases} D^{\mu}u_1(x) - \displaystyle\int_0^1 (x+t)u_1(t) + (x+t)u_2(t)dt = -\frac{1}{20} - \frac{x}{12} + \frac{4x^{1/4}\left(15-23x^2\right)}{15\Gamma(1/4)}, \\ D^{\mu}u_2(x) - \displaystyle\int_0^1 \sqrt{xt^2}u_1(t) - \sqrt{xt^2}u_2(t)dt = \frac{5x^3}{6} + \frac{9x^{4/3}}{2\Gamma(1/3)}, \\ u_1(0) = 0, u_2(0) = 0, \end{cases} \tag{30}$$

where the exact solution $U(x) = (x - x^3, x^2 - x)^T$. The numerical results are given in Tables 1 and 2, and the absolute errors of Example 1 for $m = 10, n = 3, \mu = 3/4, \alpha = \beta = 1/2$ are plotted in Figures 5 and 6. Comparisons are made between the approximate and the exact solution for $m = 10$, $n = 3, \mu = 3/4, \alpha = \beta = 1/2$ in Figures 7 and 8. Errors of u_1 and u_2 for $m = 10, n = 3, \mu = 3/4, \alpha = \beta = 1/2$ are plotted in Figures 9 and 10.

Example 2. We consider the following linear integrodifferential equations of fractional order [5]:

$$\begin{cases} D^{\mu}u_1(x) - \displaystyle\int_0^1 2xtu_1(t) + 2xtu_2(x,t)dt = \frac{83x}{80} + \frac{25x^{6/5}(11+15x)}{33\Gamma(1/5)}, \\ D^{\mu}u_2(x) - \displaystyle\int_0^1 (x+t)u_1(t) - (x+t)u_2(x,t)dt = \frac{5x^3}{6} + \frac{9x^{4/3}}{2\Gamma(1/3)}, \\ u_1(0) = 0, u_2(0) = 0, \end{cases} \tag{31}$$

where the exact solution is $U(x) = (x^3 - x^2, (15/8)x^2)^T$. We obtain the numerical results which are given in Tables 3 and 4, and the absolute errors of Example 2 for $m = 10$, $n = 3, \mu = 4/5, \alpha = \beta = 1/2$ are plotted in Figures 11 and 12. Comparisons are made between the approximate and exact solutions for $m = 10, n = 3, \mu = 4/5, \alpha = \beta = 1/2$ in Figures 13

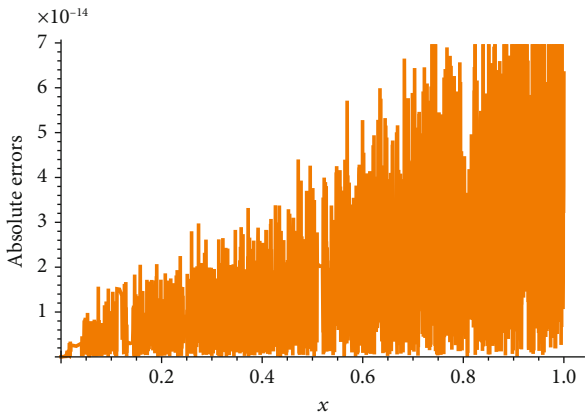

Figure 5: Absolute errors of u_1 obtained by the present method for Example 1.

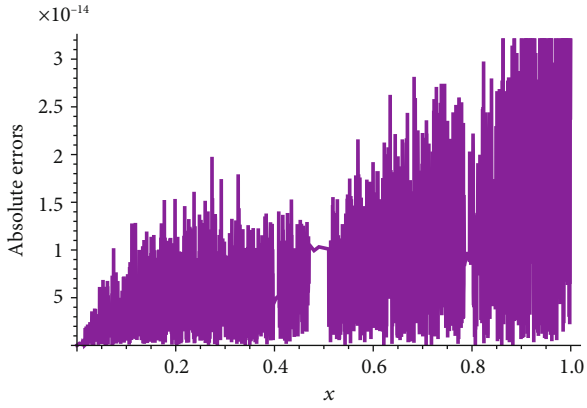

Figure 6: Absolute errors of u_2 obtained by the present method for Example 1.

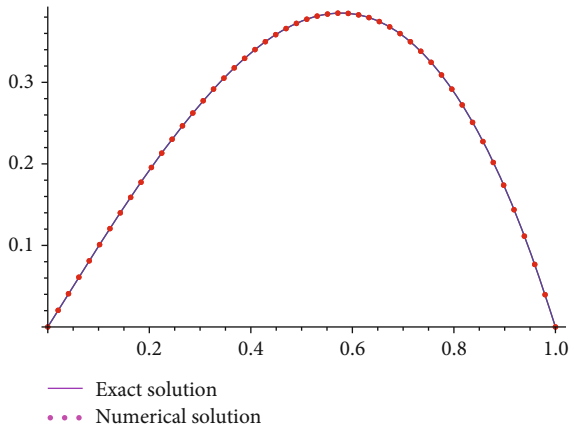

Figure 7: Comparisons between numerical and exact solutions of u_1 obtained by the present method for Example 1.

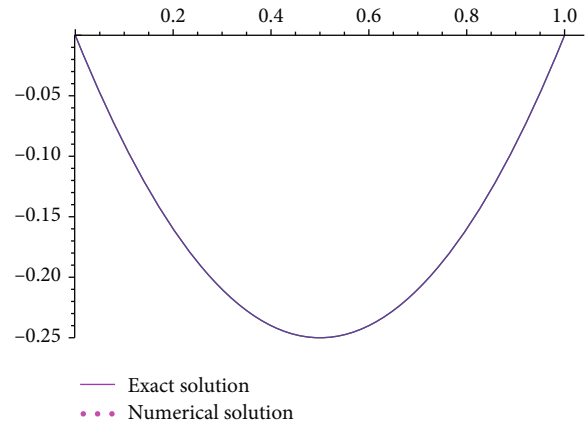

Figure 8: Comparisons between numerical and exact solutions of u_2 obtained by the present method for Example 1.

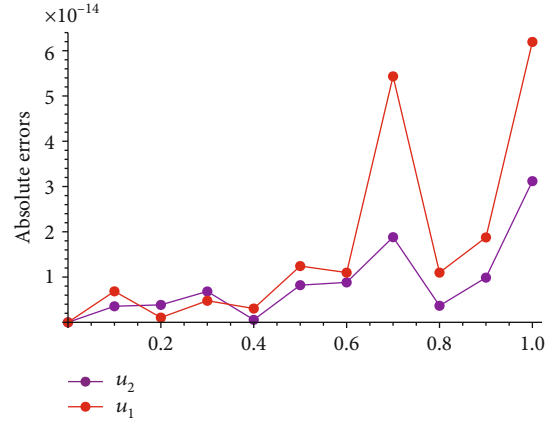

Figure 9: Absolute errors of u_1 and u_2 obtained by the present method for Example 1.

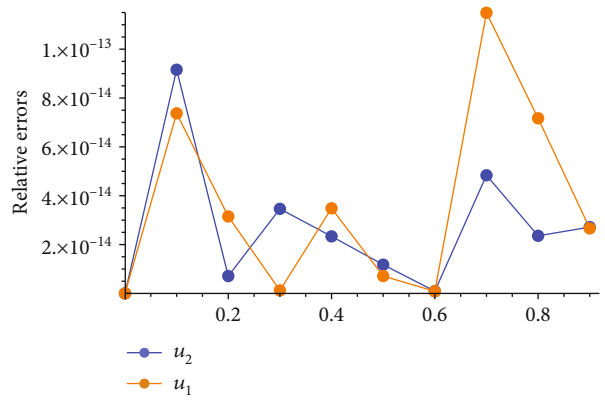

Figure 10: Relative errors of u_1 and u_2 obtained by the present method for Example 1.

Table 3: The numerical solutions of $u_1(x)$ for Example 2.

x	Exact solution	Approximate solution	$\lvert u_1(x) - u_{1,10}(x)\rvert$
0.1	−0.009	−0.009	$3.54577E-15$
0.2	−0.032	−0.032	$2.24126E-15$
0.3	−0.063	−0.063	$4.76008E-15$
0.4	−0.096	−0.096	$2.56739E-15$
0.5	−0.125	−0.125	$1.11022E-16$
0.6	−0.144	−0.144	$2.19269E-15$
0.7	−0.147	−0.147	$2.08167E-14$
0.8	−0.128	−0.128	$1.69864E-14$
0.9	−0.081	−0.081	$1.88738E-15$

Table 4: The numerical solutions of $u_2(x)$ for Example 2.

x	Exact solution	Approximate solution	$\lvert u_2(x) - u_{2,10}(x)\rvert$
0.1	0.01875	0.01875	$2.26208E-15$
0.2	0.07500	0.07500	$3.88578E-16$
0.3	0.16875	0.16875	$4.85723E-15$
0.4	0.30000	0.30000	$5.21805E-15$
0.5	0.46875	0.46875	$2.22045E-16$
0.6	0.67500	0.67500	$1.44329E-15$
0.7	0.91875	0.91875	$3.03091E-14$
0.8	1.20000	1.20000	$2.75335E-15$
0.9	1.51875	1.51875	$2.88658E-15$
1.0	1.87500	1.87500	$2.48690E-14$

and 14. Absolute errors of u_1 for $m = 10, n = 3, \mu = 3/4$ are plotted in Figure 15. Absolute errors of u_2 for $m = 10, n = 3$, $\mu = 3/4$ are showed in Figure 16.

Example 3. We consider the following linear integrodifferential equations of fractional order [4].

$$\begin{cases} D^{\mu}u_1(x) - \displaystyle\int_0^t (x-t)u_1(t) + (x-t)u_2(x,t)dt = f_1(t), \\ D^{\mu}u_2(x) - \displaystyle\int_0^t (x-t)u_1(t) - (x-t)u_2(x,t)dt = f_2(t), \\ u_1(0) = 0, u_2(0) = 0, \end{cases} \tag{32}$$

(i) where $f_1(t) = 1 + x - (t^3/3), f_2(t) = 1 - t - (t^4/12)$, the exact solution is $U(x) = (x + (x^2/2), x - (x^2/2))^T$. By the proposed algorithm, we obtain the numerical results which are given in Tables 5 and 6, and the absolute errors of this example for $m = 10, n = 2$, $\mu = 0.5, \alpha = \beta = 0$ are plotted in Figures 17 and 18. Comparisons are made between the approximate and exact solutions in Figures 19 and 20. When taking different values of α, β, errors of u_1 and u_1 for $m = 10, n = 2, \mu = 0.5$ are plotted in Figures 21–24. Figures 25 and 26 illustrate the approximate solutions of u_1 and u_2 using the present method for different values of μ which are in agreement with the exact solution. Figures 27 and 28 illustrate the approximate solutions for different values of μ compared with the exact solution in Ref. [4]

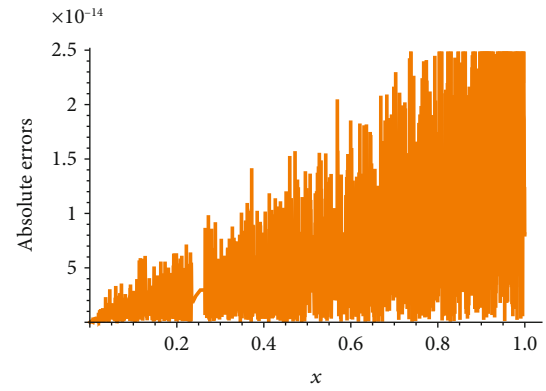

Figure 11: Absolute errors of u_1 obtained by the present method with $\alpha = \beta = 1/2$ for Example 2.

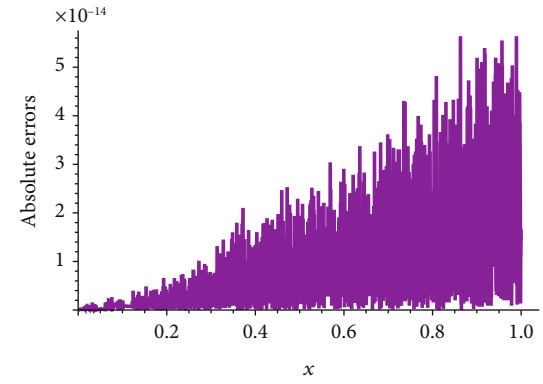

Figure 12: Absolute errors of u_2 obtained by the present method with $\alpha = \beta = 1/2$ for Example 2.

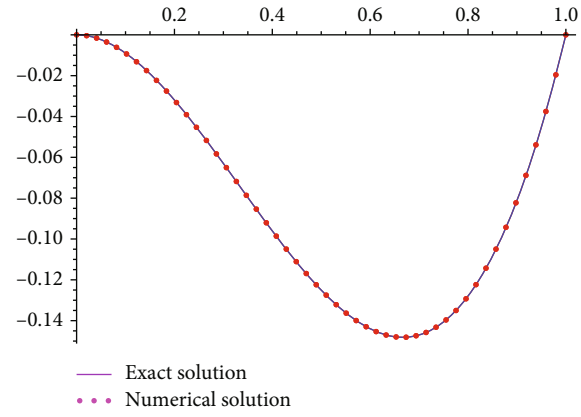

Figure 13: Comparisons between numerical and exact solutions of u_1 obtained by the present method for Example 2.

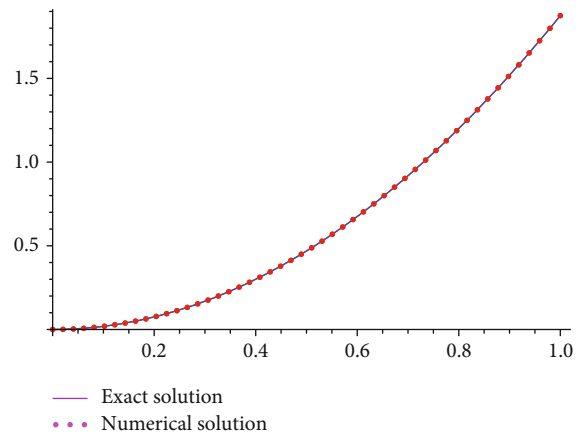

Figure 14: Comparisons between numerical and exact solutions of u_2 obtained by the present method for Example 2.

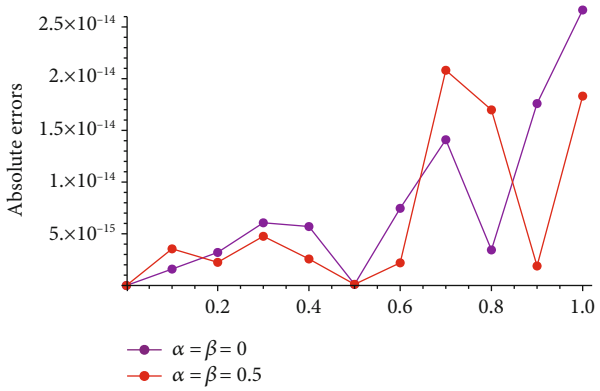

Figure 15: Absolute errors of u_1 obtained by the present method with $\alpha = \beta = 1/2$ (red), $\alpha = \beta = 0$ (purple) for Example 2.

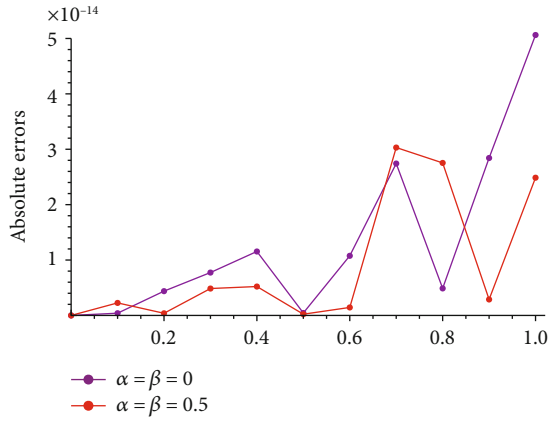

Figure 16: Absolute errors of u_2 obtained by the present method with $\alpha = \beta = 1/2$ (red), $\alpha = \beta = 0$ (purple) for Example 2.

Table 5: Comparison of the numerical result of $u_1(x)$ in Example 3.

x	Exact solution	Present method approximate solution	Present method $\lvert u_1(x) - u_{1,10}(x)\rvert$	Ref. [4] approximate solution
0.1	0.105	0.105	$4.16334E-17$	0.332
0.2	0.220	0.220	$2.22045E-16$	0.492
0.3	0.345	0.345	0.0000	0.630
0.4	0.480	0.480	$1.66533E-16$	0.759
0.5	0.625	0.625	$1.11022E-16$	0.884
0.6	0.780	0.780	$6.66134E-16$	1.007
0.7	0.945	0.945	$6.66134E-16$	1.129
0.8	1.120	1.120	$1.77636E-15$	1.252
0.9	1.305	1.305	$2.22045E-15$	1.376
1.0	1.500	1.500	$2.22045E-15$	1.500

Table 6: Comparison of the numerical result of $u_2(x)$ in Example 3.

x	Exact solution	Present method approximate solution	Present method $\lvert u_2(x) - u_{2,10}(x)\rvert$	Ref. [4] approximate solution
0.1	0.095	0.095	$9.71445E-17$	0.300
0.2	0.180	0.180	$2.49800E-16$	0.402
0.3	0.255	0.255	$3.33067E-16$	0.466
0.4	0.320	0.320	$3.88578E-16$	0.506
0.5	0.375	0.375	$1.11022E-16$	0.530
0.6	0.420	0.420	$2.22045E-16$	0.542
0.7	0.455	0.455	$3.88578E-16$	0.544
0.8	0.480	0.480	$6.10623E-16$	0.537
0.9	0.495	0.495	$7.77156E-16$	0.522
1.0	0.500	0.500	$1.11022E-15$	0.500

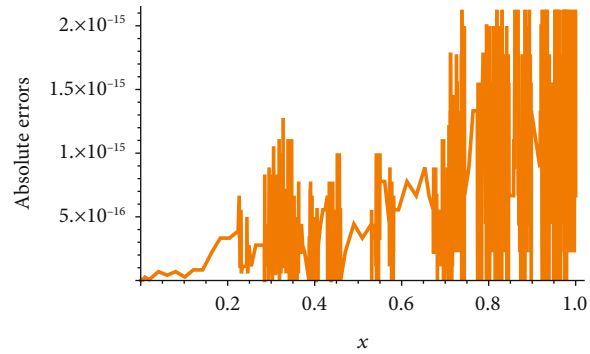

Figure 17: Absolute errors of u_1 obtained by the present method for Example 3.

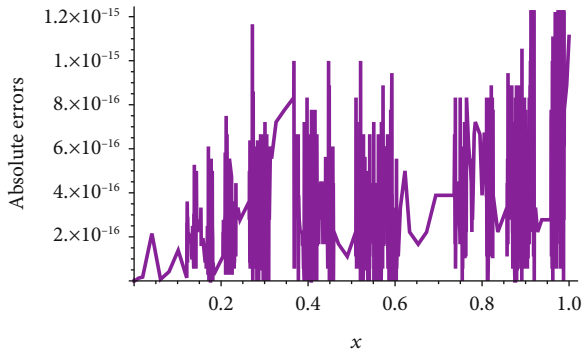

Figure 18: Absolute errors of u_2 obtained by the present method for Example 3.

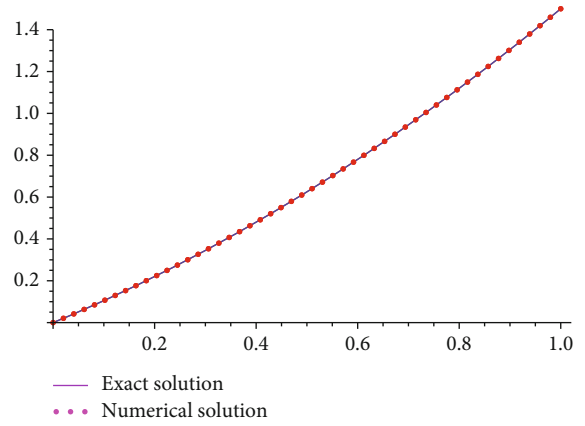

Figure 19: Comparisons between numerical and exact solutions of u_1 obtained by the present method for Example 3.

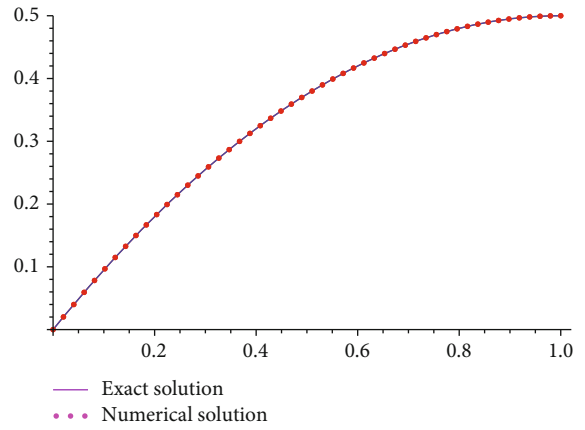

Figure 20: Comparisons between numerical and exact solutions of u_2 obtained by the present method for Example 3.

(ii) where $f_1(t) = (0.506344 - 1.86957x^{1.5} + 1.81521x^{1.7})/x^{0.5}$, $f_2(t) = 1.81521(-0.239522 - 0.42377x + x^{1.2})$, the exact solution is $U(x) = (x^{0.3} + x^2, x^2)^T$. The absolute errors of this example for $m = 200, n = 2$, $\mu = 0.2, \alpha = \beta = 0.5$ are given in Figures 29 and 30

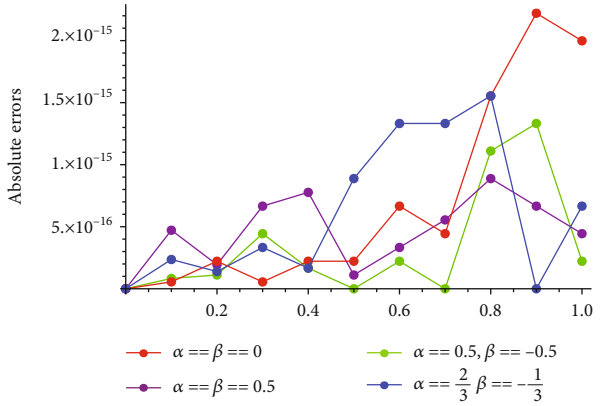

Figure 21: Absolute errors of u_1 obtained by the present method for Example 3.

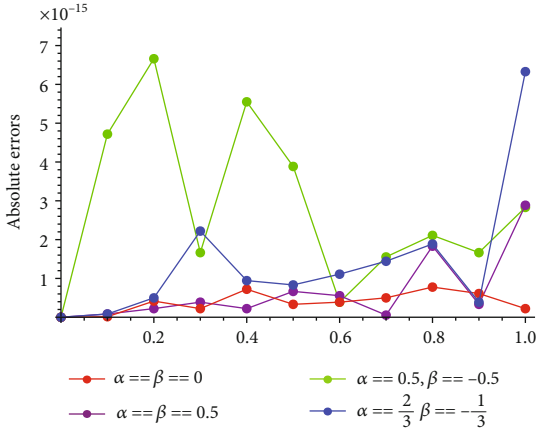

Figure 22: Absolute errors of u_2 obtained by the present method for Example 3.

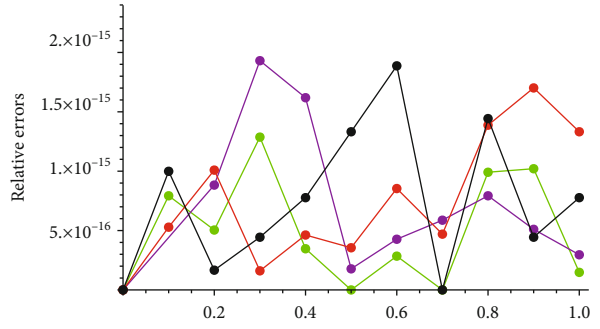

Figure 23: Relative errors of u_1 obtained by the present method with $\alpha = \beta = 0$ (red), $\alpha = \beta = 0.5$ (purple), $\alpha = 0.5, \beta = -0.5$ (green), and $\alpha = 2/3, \beta = -1/3$ (black) for Example 3.

5. Conclusions and Remarks

In this paper, linear integrodifferential equations of fractional order have been solved by the reproducing kernel interpolation collocation method with reproducing kernel function in the form of Jacobi polynomials for the first time. Comparisons are made between the approximate and exact solutions. We verify the feasibility of this method by selecting different

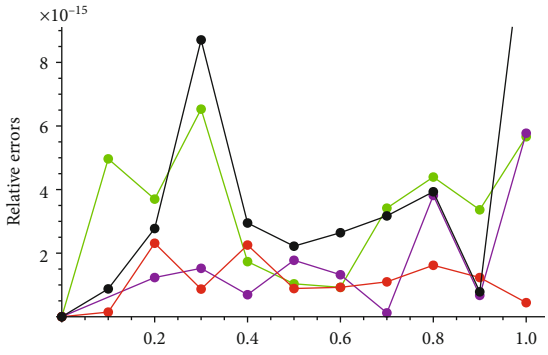

Figure 24: Relative errors of u_2 obtained by the present method with $\alpha = \beta = 0$ (red), $\alpha = \beta = 0.5$ (purple), $\alpha = 0.5$, $\beta = -0.5$ (green), and $\alpha = 2/3$, $\beta = -1/3$ (black) for Example 3.

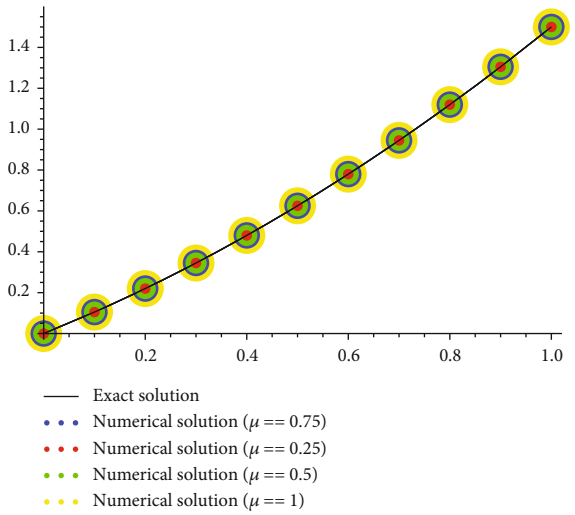

Figure 25: The numerical solution of u_1 obtained by the present method with $\mu = 0.75,0.5,0.25,1$ and exact solutions for Example 3.

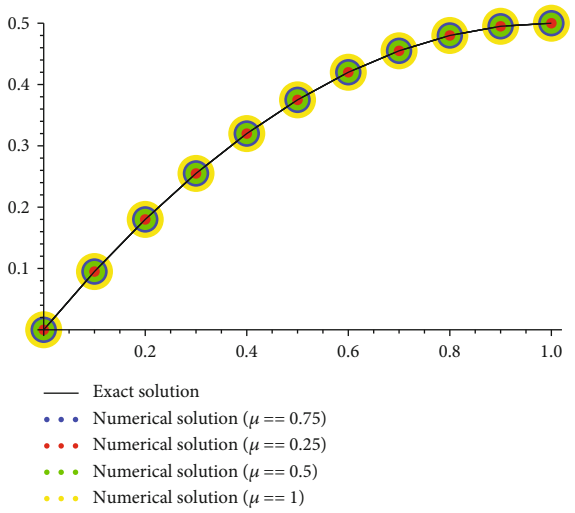

Figure 26: The numerical solution of u_2 obtained by the present method with $\mu = 0.75,0.5,0.25,1$ and exact solutions for Example 3.

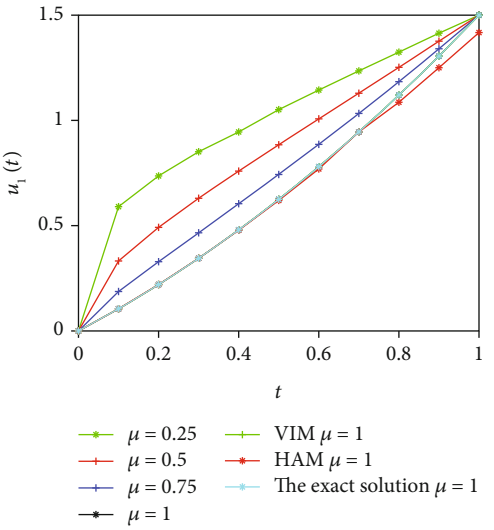

Figure 27: The numerical solution of u_1 with $\mu = 0.75,0.5,0.25,1$ for Example 3 in Ref. [4].

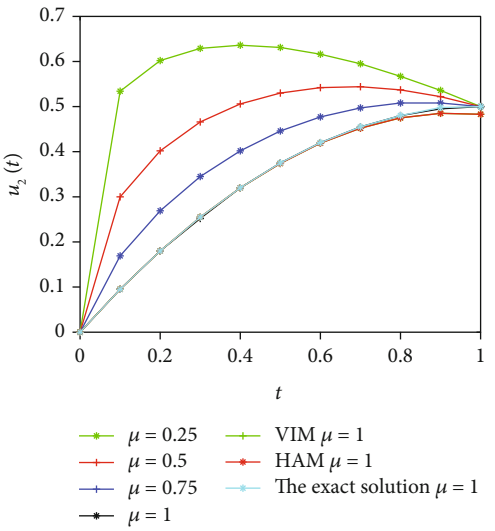

Figure 28: The numerical solution of u_2 with $\mu = 0.75,0.5,0.25,1$ for Example 3 in Ref. [4].

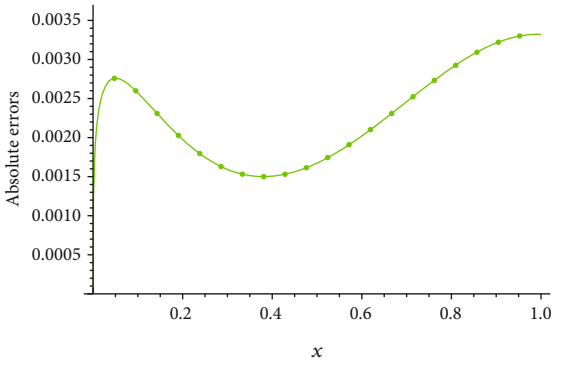

Figure 29: Absolute errors of u_1 obtained by the present method for Example 3.

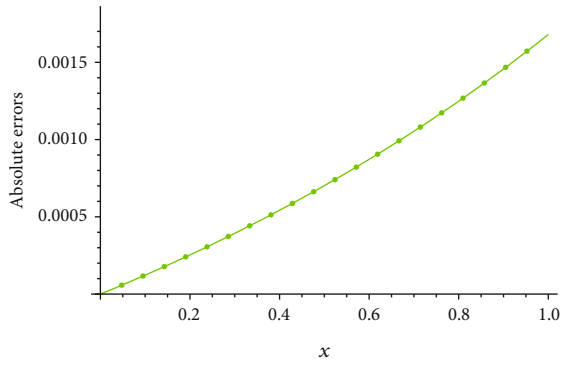

FIGURE 30: Absolute errors of u_2 obtained by the present method for Example 3.

parameters μ, α, β. From all tables and figures, we obtain that the algorithm is remarkably accurate and effective.

All computations are performed by the Mathematica 7.0 software package.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This paper is supported by the Natural Science Foundation of Inner Mongolia (2017MS0103) and the National Natural Science Foundation of China (11361037).

References

[1] D. S. Mohammed, "Numerical solution of fractional integro-differential equations by least squares method and shifted Chebyshev polynomial," *Mathematical Problems in Engineering*, vol. 2014, Article ID 431965, 965 pages, 2014.

[2] K. Kumar, R. K. Pandey, and S. Sharma, "Comparative study of three numerical schemes for fractional integro- differential equations," *Journal of Computational and Applied Mathematics*, vol. 315, pp. 287–302, 2017.

[3] F. A. Hendi, W. Shammakh, and H. Al-badrani, "Existence result and approximate solutions for quadratic integro-differential equations of fractional order," *Journal of King Saud University-Science*, vol. 31, no. 3, pp. 314–321, 2019.

[4] O. H. Mohammed and A. M. Malik, "A modified computational algorithm for solving systems of linear integro- differential equations of fractional order," *Journal of King Saud University-Science*, vol. 31, no. 4, pp. 946–955, 2019.

[5] A. M. S. Mahdy, "Numerical studies for solving fractional integro-differential equations," *Journal of Ocean Engineering and Science*, vol. 3, no. 2, pp. 127–132, 2018.

[6] Z.-Y. Li, Y.-L. Wang, F.-G. Tan, X.-H. Wan, H. Yu, and J.-S. Duan, "Solving a class of linear nonlocal boundary value problems using the reproducing kernel," *Applied Mathematics and Computation*, vol. 265, pp. 1098–1105, 2015.

[7] X. Y. Li and B. Y. Wu, "Error estimation for the reproducing kernel method to solve linear boundary value problems," *Journal of Computational and Applied Mathematics*, vol. 243, pp. 10–15, 2013.

[8] Y. L. Wang, Z. Y. Li, Y. Cao, and X. H. Wan, "A new method for solving a class of mixed boundary value problems with singular coefficient," *Applied Mathematics and Computation*, vol. 217, no. 6, pp. 2768–2772, 2010.

[9] Y. Wang, T. Chaolu, and Z. Chen, "Using reproducing kernel for solving a class of singular weakly nonlinear boundary value problems," *International Journal of Computer Mathematics*, vol. 87, no. 2, pp. 367–380, 2010.

[10] Y. Wang, M. Du, F. Tan, Z. Li, and T. Nie, "Using reproducing kernel for solving a class of fractional partial differential equation with non-classical conditions," *Applied Mathematics and Computation*, vol. 219, no. 11, pp. 5918–5925, 2013.

[11] Y. L. Wang, L. N. Jia, and H. L. Zhang, "Numerical solution for a class of space-time fractional equation by the piecewise reproducing kernel method," *International Journal of Computer Mathematics*, vol. 96, no. 10, pp. 2100–2111, 2019.

[12] R. Ketabchi, R. Mokhtari, and E. Babolian, "Some error estimates for solving Volterra integral equations by using the reproducing kernel method," *Journal of Computational and Applied Mathematics*, vol. 273, pp. 245–250, 2015.

[13] O. A. Arqub, M. Al-Smadi, S. Momani, and T. Hayat, "Application of reproducing kernel algorithm for solving second-order, two-point fuzzy boundary value problems," *Soft Computing*, vol. 21, no. 23, pp. 7191–7206, 2017.

[14] H. Sahihi, S. Abbasbandy, and T. Allahviranloo, "Reproducing kernel method for solving singularly perturbed differential-difference equations with boundary layer behavior in Hilbert space," *Journal of Computational and Applied Mathematics*, vol. 328, pp. 30–43, 2018.

[15] Y. L. Wang, C. L. Temuer, and J. Pang, "New algorithm for second-order boundary value problems of integro-differential equation," *Journal of Computational and Applied Mathematics*, vol. 229, no. 1, pp. 1–6, 2009.

[16] E. H. Doha, A. H. Bhrawy, and S. S. Ezz-Eldien, "A new Jacobi operational matrix: an application for solving fractional differential equations," *Applied Mathematical Modelling*, vol. 36, no. 10, pp. 4931–4943, 2012.

[17] M. Khaleghi, E. Babolian, and S. Abbasbandy, "Chebyshev reproducing kernel method: application to two-point boundary value problems," *Advances in Difference Equations*, vol. 2017, no. 1, Article ID 26, 2017.

[18] B. Y. Wu and Y. Z. Lin, *Application of the Reproducing Kernel Space*, Science Press, 2012.

[19] X. Li and B. Wu, "A new reproducing kernel method for variable order fractional boundary value problems for functional differential equations," *Journal of Computational and Applied Mathematics*, vol. 311, pp. 387–393, 2017.

[20] X. Y. Li and B. Y. Wu, "A new reproducing kernel collocation method for nonlocal fractional boundary value problems with non-smooth solutions," *Applied Mathematics Letters*, vol. 86, pp. 194–199, 2018.

[21] F. Geng and M. Cui, "A reproducing kernel method for solving nonlocal fractional boundary value problems," *Applied Mathematics Letters*, vol. 25, no. 5, pp. 818–823, 2012.

[22] F. Z. Geng and S. P. Qian, "An optimal reproducing kernel method for linear nonlocal boundary value problems," *Applied Mathematics Letters*, vol. 77, pp. 49–56, 2018.

[23] W. Jiang and Z. Chen, "Solving a system of linear Volterra integral equations using the new reproducing kernel method," *Applied Mathematics and Computation*, vol. 219, no. 20, pp. 10225–10230, 2013.

[24] X. Li, H. Li, and B. Wu, "Piecewise reproducing kernel method for linear impulsive delay differential equations with piecewise constant arguments," *Applied Mathematics and Computation*, vol. 349, pp. 304–313, 2019.

[25] F. Z. Geng and Z. Q. Tang, "Piecewise shooting reproducing kernel method for linear singularly perturbed boundary value problems," *Applied Mathematics Letters*, vol. 62, pp. 1–8, 2016.