Journal of Fractional Calculus and Applications
Vol. 13(1) Jan. 2022, pp. 171-188
ISSN: 2090-5858.
http://math-frac.org/Journals/JFCA/

A NEW OPERATIONAL MATRIX OF INTEGRATION BASED ON THE INDEPENDENCE POLYNOMIAL OF GRAPH TO SOLVE FRACTIONAL POISSON EQUATION

CHANDRALI BAISHYA

ABSTRACT. In this paper, we have derived a new operational matrix of fractional integration by using the independence polynomial of a complete bipartite graph and applied it to solve the Poisson equation with Dirichlet boundary conditions. While deriving the operational matrix, the Caputo sense fractional derivatives are considered. Series solutions are found by using the collocation matrix method. The main characteristic of the approach is that it reduces a complex fractional differential equation to a system of algebraic equations. The error bound and computational complexity of the projected algorithm are also investigated. Solutions obtained for the Poisson equation have established the relevance and applicability of the method described. Comparative analysis of the obtained results with the exact solutions convinces that the present method can be considered an efficient numerical tool for solving fractional differential equations.

1. INTRODUCTION

The Poisson equation is the famous elliptic partial differential equation that governs many physical phenomena such as steady heat conduction, seepage through porous media, the irrotational flow of an ideal fluid, distribution of electrical and magnetic potential, torsion of the prismatic shaft, bending of prismatic beams, distribution of gravitational potentials and many others. It has the general form:

$$\bigtriangledown^2 w = g. \tag{1}$$

where g is the source function defined in the domain Ω and $\Omega \in \mathbb{R}^2$ or $\Omega \in \mathbb{R}^3$. The boundary conditions on the boundary $\partial\Omega$ for this equation can be Dirichlet boundary condition or Neumann boundary condition or mixed boundary condition [1].

In recent years, fractional calculus and its theory have been remarkably developed [2, 3, 4, 5] and are extensively used in the different fields of science and technology. Pudlobny [4] and Kilbas et al. [5] have investigated the existence and uniqueness of solutions to the fractional differential equations (FDEs). The utility of fractional partial differential equations (FPDEs) in mathematical modelling has attracted the attention of many scientists

2010 *Mathematics Subject Classification.* 35J15, 65M70, 35R11.

Key words and phrases. Caputo fractional derivative, complete bipartite graph, fractional Poisson equation, operational matrix.

Submitted Dec. 11, 2020. Revised June 24, 2021.

in recent years. Kumar et al. [6] and Veeresh et al.[7] analyzed Fornberg– Whitham equation and Fisher-Kolmogorov equation respectively incorporating fractional derivative with Mittag–Leffler type kernel. Fluid mechanics [8], the nonlinear oscillation of earthquakes [9], traffic model [10], viscoelasticity [11], continuum and statistical mechanics [12], solid mechanics [13], economics [14, 15], mathematical biology [16, 17, 18], bioengineering [19], dynamics of interfaces between nanoparticles and substrates [20] are some of such examples where FDEs and in particular FPDEs arise.

There are many effective numerical techniques to solve FPDEs viz. homotopy perturbation method [21], homotopy analysis method [22], q-homotopy analysis method [23, 7], adomian decomposition method [24], differential transform method [25], the variational iteration method [26], the heat-balance integral method [27] and others [28, 29, 30, 31, 32, 33]. Recently Chen et al. have proposed Kansa method which belongs to the RBF collocation method for solving fractional diffusion equations [34]. Wasim et al. [35] have used a B-Spline Collocation method for solving the generalized Burgers-Fisher and Burgers-Huxley equations. In [36], Sayevand et al. applied B-Spline collocation method to solve fractional diffusion equations in transport dynamic systems. In [37], Sun et al. applied a novel finite difference method for solving variable-order time fractional diffusion equation. Fu and Yang [38] have proposed the Laplace transformed boundary particle method for solving time fractional diffusion equations.

Because of the ability to represent a function at different levels of resolution, orthogonal polynomial bases remain a popularly adopted trend for numerical solutions of both integer order and fractional order partial differential equations. In recent years, we can observe many research publications involving orthogonal polynomial-based numerical methods to solve FPDEs. In this context, researchers have solved FPDEs by generating operational matrix of fractional derivatives using orthogonal polynomials such as Legendre polynomial [39, 40], Laguerre polynomial [41], Jacobi polynomial [42, 43, 44], Chebyshev polynomial [45, 46], Genocci polynomial [47, 48].

In the same line of curiosity to invent new methods for FDEs, few mathematicians also developed new numerical techniques involving non-orthogonal bases and achieved remarkable accuracy. Bernstein polynomial is one of the non-orthogonal polynomials widely used to solve integer order and fractional order ordinary and partial differential equations. Parand et al. [49] have introduced an operation matrix method based on Bernstein polynomials for the Riccati differential equation and Volterra population model. Rostamy et al. [50] treated FDEs by an operational matrix of fractional derivatives derived from Bernstein polynomial. Hossein and Haleh [51] have solved fractional order optimal control problems by operation matrix method based on Bernstein polynomials. Numerical solutions of time-fractional order telegraph equations are solved by Bernstein polynomials operational matrices by Asgari et al. [52]. Recently, some of the researchers have applied polynomials derived from various graphs to solve linear and nonlinear ordinary differential equations, partial differential equations, integral equations, and integro-differential equations [53, 54, 55, 56, 57]. Ramane et al. [53] have solved integral equations using Hosoya polynomial of a path graph. The Fibonacci polynomial being one of the most fascinating polynomials with their many generalized properties, attracted applied mathematicians to use them in solving differential equations. While articles using these polynomials in different applications are limited, a collocation procedure was built in [58, 59] for treating BVPs using the Fibonacci operational matrix of derivatives. Waleed and Youssri [60] solved FDEs using an operational matrix calculated from Fibonacci polynomials.

The major goal of this research is to contribute a novel solution technique for dealing with FDEs. Even though numerous orthogonal polynomials are utilized to solve FDEs by producing operational matrices, non-orthogonal polynomials are rarely used in this direction with the same enthusiasm. Graph theory is a large field of studies in which academics are always working to contribute new theories connected to diverse physical structures in graphical form, resulting in many new polynomials in this subject. This motivates us to formulate a novel numerical approach by using polynomial related to a graph. We created an operational matrix of fractional integration in this study based on the independence polynomial of a complete bipartite graph. We demonstrated the use of this operational matrix for converting the FPDEs to a system of algebraic equations by taking the Poisson equation as a case study. The proposed technique is a first step in developing a new numerical strategy for dealing with FDEs. The key feature of the technique is that the polynomial under consideration is non-orthogonal, and the analysis reveals that the algorithm produces an efficient operational matrix from a non-orthogonal polynomial in a straightforward manner. The proposed scheme converts complex differential equations to a system of algebraic equations. The strength of the suggested algorithm is determined by its error-bound analysis and computational complexity study. The derived operational matrix could give good accuracy for the FDEs irrespective of the choice of collocation points. One notable observation is that the algorithm needs more CPU time if the number of nodal points is large. In this paper, we consider the Poisson equation with fractional derivatives in the Caputo sense in the domain $\Omega = \{(x,y) : 0 \leq x \leq a, 0 \leq x \leq b\}$, a and b are finite constant, concerning the solution $w(x,y)$ satisfying the equation

$$\frac{\partial^{\alpha} w(x,y)}{\partial x^{\alpha}} + \frac{\partial^{\beta} w(x,y)}{\partial y^{\beta}} = g(x,y), \quad 1 < \alpha, \beta \leq 2, \, \text{for}\, (x,y) \,\text{in}\, \Omega, \tag{2}$$

with boundary conditions

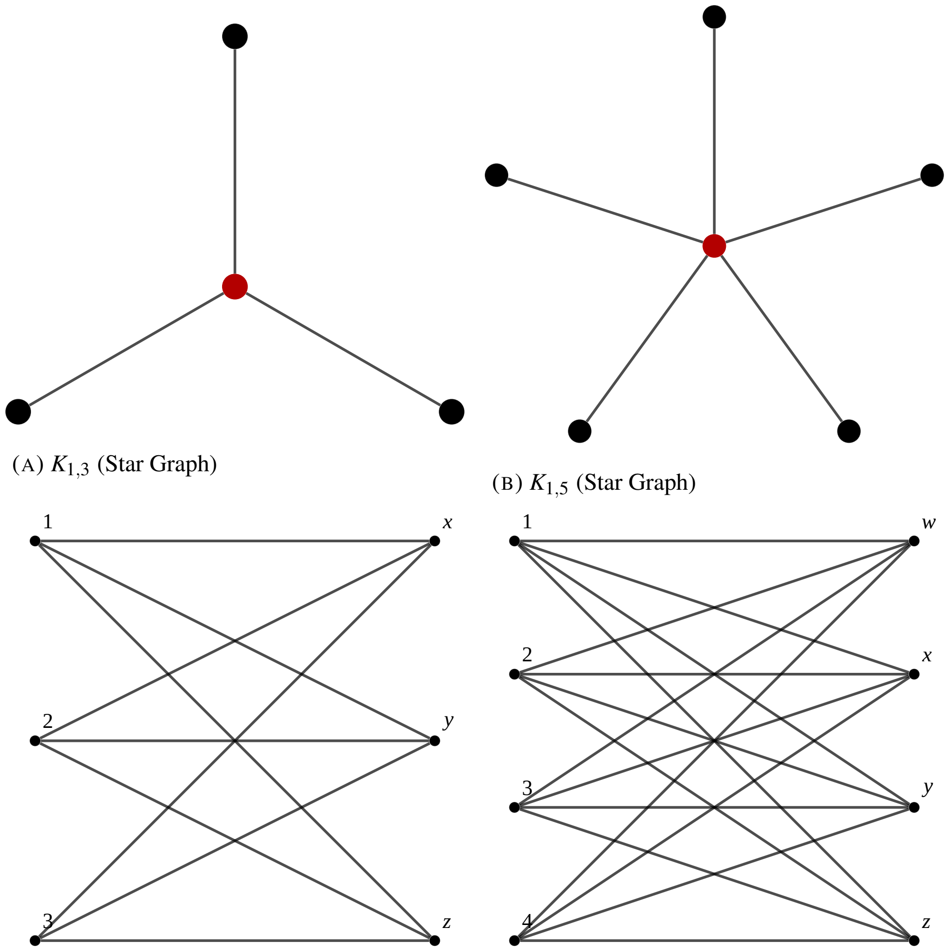
$$w(x,y) = \psi(x,y), \, \text{for}\, (x,y) \,\text{on the boundary}\, \partial\Omega. \tag{3}$$

where $\partial\Omega$ is the boundary of the region Ω.

The paper is organized as follows: In Section 2, we give the definitions related to the independence polynomial of a complete bipartite graph and some preliminaries of fractional calculus. In Section 3, we have described the approximation of a function by the independence polynomial of a complete bipartite graph. Construction of operational matrix of fractional integration by the independence polynomial of a complete bipartite graph, error bound, and computational complexity analysis of the algorithm is presented in Section 4. In Section 5, we have demonstrated the method of solving the Poisson equation by using the complete bipartite independence polynomial collocation (CBIPC) method. To investigate the efficiency of the present method numerically, in Section 6, we have solved three Poisson equation examples involving Caputo fractional derivative. Conclusion is presented in Section 7.

2. Independence polynomial and preliminaries

2.1. **Independence polynomial of a complete bipartite graph.** A simple graph is a pair $G = (V,E)$, where V is a set of elements called vertices, and E is a set of elements called edges. Let $u, v \in V$. The vertices u and v are said to be adjacent if there is an edge between u and v. An independent set in the graph G is a set of pairwise non-adjacent vertices. The independence number denoted by $\alpha(G)$ is the cardinality of a maximum independent set of G. We have referred to the literature [61], to get the theoretical concept of independence



(A) $K_{1,3}$ (Star Graph)

(B) $K_{1,5}$ (Star Graph)

(C) $K_{3,3}$ (Utility Graph)

(D) $K_{4,4}$ (Caley Graph)

FIGURE 1. Class of complete bipartite graphs.

polynomial. The independence polynomial of a graph G is the polynomial whose coefficient on x^k is given by the number of independent sets of order k in G. We denote this polynomial by $I(G;x)$.
So, $I(G;x) = \sum_{k=1}^{\alpha(G)} c_k x^k$, where x^k is the number of independent sets of order k in G.
A bipartite graph is a graph whose vertices can be partitioned into two independent sets A and B such that every edge connects a vertex in A to one in B. The Complete Bipartite graph, denoted by $K_{m,n}$, is a bipartite graph on $m+n$ vertices. The vertices in $K_{m,n}$ are partitioned into two independent sets A and B, where $|A| = m$ and $|B| = n$. Additionally, every vertex in A is adjacent to every vertex in B. Star graph, Utility graph, Cayley graph are some of the well known complete bipartite graphs shown in the figure 1.

In this paper, we shall assume that $m = n$. The recurrence relation for $I(K_{n,n};x)$ is

$$I(K_{n,n};x) = I(K_{n-1,n-1};x) + 2x(1+x)^{n-1} \quad with \quad I(K_{0,0}) = 1. \tag{4}$$

The recurrence relation (4) gives closed form of the independence polynomial of a complete bipartite graph as

$$I(K_{n,n}) = 2(1+x)^n - 1. \tag{5}$$

2.2. **Preliminaries.**

Definition 1. The Riemann–Liouville fractional order integral operator is defined by [4]

$$J_x^{\alpha} f(x) = \frac{1}{\Gamma(\alpha)} \int_0^x \frac{f(t)}{(x-t)^{\alpha-1}} dt, \, \alpha > 0.$$
$$J^0 f(x) = f(x).$$

For Riemann–Liouville fractional order integral operator, we have

$$J_x^{\alpha} x^n = \frac{\Gamma(n+1)}{\Gamma(n+1+\alpha)} x^{n+\alpha}. \tag{6}$$

3. Function Approximation

A function $g(x,y) \in L^2([0,1]\times[0,1])$ can be expressed in terms of the independence polynomials of a complete bipartite graph basis as

$$\begin{aligned} g(x,y) &= \sum_{i=0}^{m}\sum_{j=0}^{m} g_{ij} I(K_{ii;x}) I(K_{jj;y}) \\ &= \Psi^T(x) G \Psi(y). \end{aligned}$$

where $\Psi^T(x) = [I(K_{00;x}), I(K_{11;x}), \ldots I(K_{mm;x})]$.

$$G = \begin{pmatrix} g_{00} & g_{01} & \cdots & g_{0m} \\ g_{10} & g_{11} & \cdots & g_{1m} \\ . & . & \cdots & . \\ . & . & \cdots & . \\ g_{m0} & g_{m1} & \cdots & g_{mm} \end{pmatrix}$$

G is computed by the equation

$$G = Q^{-1} < \Psi(x), < \Psi(y), g(x,y) >> Q^{-1}, \tag{7}$$

where

$$Q =< \Psi(x), \Psi(x) >= \int_0^1 \Psi(x)\Psi^T(x) dx. \tag{8}$$

4. Generlized complete bipartite operational matrix of fractional integration

Since the order of fractional derivatives α and β in equation (2) satisfy $1 < \alpha, \beta \leq 2$, in the process of integration we encounter first derivative of $\Psi(x)$. The first derivative of the vector $\Psi(x)$ for $m = 6$ can be written as:

$$\frac{d}{dx}\Psi(x) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 3 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 4 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 & 5 & 0 & 0 \\ 6 & 0 & 0 & 0 & 0 & 6 & 0 \end{pmatrix} \Psi(x).$$

Therefor, in general we may write first derivative of the vector $\Psi(x)$ as

$$\frac{d}{dx}\Psi(x) = D^{(1)}\Psi(x). \tag{9}$$

where $D^{(1)} = (d_{ij})$ is the complete bipartite operational matrix of derivatives of order $(m+1)\times(m+1)$ and

$$d_{ij} = \begin{cases} 2, & \text{for } i = 1, j = 0, \\ i, & \text{for } j = 0 \text{ and } j = i-1, \\ 0, & \text{otherwise.} \end{cases}$$

By using relation (9) it is clear that

$$\frac{d^n\psi(x)}{dx^n} = (D^{(1)})^n\Psi(x) \quad \text{where } n \in \mathbb{N}.$$

The following theorem generalizes the operational matrix of fractional integration based on the independence polynomial of a complete bipartite graph.

Theorem 1. Let $\Psi(x)$ be the complete bipartite vector defined in (5) and let $\alpha > 0$ then

$$J_x^\alpha\Psi(x) \approx x^\alpha J^{(\alpha)}\Psi(x).$$

where $J_x^{(\alpha)}$ is the $(m+1)\times(m+1)$ operational matrix of fractional integration of order α and is defined as follows:

$$J^{(\alpha)} = \begin{pmatrix} \Theta^\alpha_{0,0} & 0 & 0 & \dots & 0 & \dots & 0 \\ \Theta^\alpha_{1,0} & \Theta^\alpha_{1,1} & 0 & \dots & 0 & \dots & 0 \\ \Theta^\alpha_{2,0} & \Theta^\alpha_{2,1} & \Theta^\alpha_{2,2} & \dots & 0 & \dots & 0 \\ . & . & . & & . & & \\ . & . & . & & . & & \\ . & . & . & & . & & \\ \Theta^\alpha_{i,0} & \Theta^\alpha_{i,1} & \Theta^\alpha_{i,2} & \dots & \Theta^\alpha_{i,i} & \dots & 0 \\ . & . & . & & . & & \\ . & . & . & & . & & \\ . & . & . & & . & & \\ \Theta^\alpha_{m,0} & \Theta^\alpha_{m,1} & \Theta^\alpha_{m,2} & \dots & \Theta^\alpha_{m,i} & \dots & \Theta^\alpha_{m,m} \end{pmatrix} \tag{10}$$

Proof. We have

$$I(K_{ii};x) = 2\sum_{k=0}^{i}\binom{i}{k}x^k - 1 \tag{11}$$

Using equations (6) and (11) we get,

$$\begin{aligned} J_x^\alpha I(K_{ii};x) =& 2\sum_{k=0}^{i}\binom{i}{k}\frac{\Gamma(k+1)}{\Gamma(k+1+\alpha)}x^{k+\alpha} - \frac{1}{\Gamma(1+\alpha)}x^\alpha \\ =& 2x^\alpha\sum_{k=1}^{i}\frac{i!}{(i-k)!\Gamma(k+1+\alpha)}x^k + \frac{1}{\Gamma(1+\alpha)}, \quad i = 1,2,...,m. \end{aligned} \tag{12}$$

Approximating x^k by $(m+1)$ the independence polynomial of a complete bipartite graph, we have,

$$x = -\frac{1}{2}\big(I(K_{00};x) - I(K_{11};x)\big),$$

$$x^2 = \frac{1}{2}\big(I(K_{00};x) - 2I(K_{11};x) + I(K_{22};x)\big),$$

$$x^3 = -\frac{1}{2}\big(I(K_{00};x) - 3I(K_{11};x) + 3I(K_{22};x) - I(K_{33};x)\big).$$

In general

$$x^k = (-1)^k \frac{1}{2}\sum_{j=0}^{k}(-1)^j \binom{k}{j} I(K_{jj};x) = \sum_{j=0}^{k} a_{k,j} I(K_{jj};x), \quad k = 0,1,2,... \tag{13}$$

Where,

$$a_{k,j} = (-1)^{k+j}\frac{1}{2}\binom{k}{j}.$$

Using equation (13) in equation (12) we get,

$$\begin{aligned}
J_x^\alpha I(K_{ii};x) =& 2x^\alpha \sum_{k=1}^{i}\sum_{j=1}^{k} \frac{i!}{(i-k)!\Gamma(k+1+\alpha)} a_{k,j} I(K_{jj};x)\\
&+ x^\alpha \big(\sum_{k=1}^{i} \frac{i!(-1)^k}{(i-k)!\Gamma(k+1+\alpha)} + \frac{1}{\Gamma(1+\alpha)}\big) I(K_{00};x)\\
=& x^\alpha \sum_{j=1}^{i}\sum_{k=1}^{i} \frac{i!}{(i-k)!\Gamma(k+1+\alpha)} (-1)^{k+j}\binom{k}{j} I(K_{jj};x)\\
&+ x^\alpha \big(\sum_{k=1}^{i} \frac{i!(-1)^k}{(i-k)!\Gamma(k+1+\alpha)} + \frac{1}{\Gamma(1+\alpha)}\big) I(K_{00};x)\\
=& x^\alpha \big(\sum_{j=1}^{i} \Theta^\alpha_{i,j} I(K_{jj};x) + \Theta^\alpha_{i,0} I(K_{00};x)\big), \quad i = 1,2,...,m
\end{aligned} \tag{14}$$

Where, for $i = 1,2,...,m$

$$\Theta_{i,j} = \begin{cases} \sum_{k=1}^{i} \frac{i!(-1)^k}{(i-k)!\Gamma(k+1+\alpha)} + \frac{1}{\Gamma(1+\alpha)}, & \text{for } j = 0,\\ \sum_{k=1}^{i} \frac{i!}{(i-k)!\Gamma(k+1+\alpha)}(-1)^{k+j}\binom{k}{j}, & \text{for } j \le i,\\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

$$\Theta^\alpha_{0,0} = \frac{1}{\Gamma(1+\alpha)}.$$

And,

$$J_x^\alpha I(K_{00};x) = \Theta^\alpha_{0,0} I(K_{00};x). \tag{16}$$

We have,

$$J_x^\alpha I(K_{ii};x) \approx x^\alpha [\Theta^\alpha_{i,0}, \Theta^\alpha_{i,1}, ..., \Theta^\alpha_{i,m}] \psi(x), \quad i = 0,1...m. \tag{17}$$

Equations (15), (16) and (17) yield the result (10). □

Theorem 2. The error $|\Upsilon_m| = |J_x^\alpha w(x,y) - J_x^\alpha w_m(x,y)|$ in approximating $J_x^\alpha w(x,y)$ with operational matrix of fractional integration is bounded as follows:

$$|J_x^\alpha w(x,y) - J_x^\alpha w_m(x,y)| \le\le \sum_{i=m+1}^{\infty}\sum_{j=m+1}^{\infty} |W_{ij}| \left(\sum_{p=1}^{m} |\Theta^\alpha_{i,p}| \left(2\sum_{k=1}^{p}\binom{p}{k} + 1\right) + |\Theta^\alpha_{i,0}|\right) \left(2\sum_{k=0}^{j}\binom{j}{k} + 1\right),$$

where w_m is the approximation of the function $w(x)$ based on the independence polynomials a complete bipartite graph, $W_{i,j}$, $i = 0,1,2,...,m$, $j = 0,1,2,...,m$ is the coefficients matrix of this approximation and $\Theta^{\alpha}_{i,j}$ is given by equation (15).

Proof. We consider,

$$w(x,y) = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} W_{ij} I(K_{ii};x) I(K_{jj};y).$$

Taking first m terms, we get,

$$w_m(x,y) = \sum_{i=0}^{m}\sum_{j=0}^{m} W_{ij} I(K_{ii};x) I(K_{jj};y).$$

Then,

$$w(x,y) - w_m(x,y) = \sum_{i=m+1}^{\infty}\sum_{j=m+1}^{\infty} W_{ij} I(K_{ii};x) I(K_{jj};y).$$

$$\begin{aligned}
|J_x^{\alpha} w(x,y) - J_x^{\alpha} w_m(x,y)| &= |\sum_{i=m+1}^{\infty}\sum_{j=m+1}^{\infty} W_{ij} J_x^{\alpha} I(K_{ii};x) I(K_{jj};y)| \\
&= |\sum_{i=m+1}^{\infty}\sum_{j=m+1}^{\infty} W_{ij} x^{\alpha} \left(\sum_{p=1}^{m} \Theta^{\alpha}_{i,p} I(K_{pp};x) + \Theta^{\alpha}_{i,0} I(K_{00};x) \right) I(K_{jj};y)| \\
&\leq \sum_{i=m+1}^{\infty}\sum_{j=m+1}^{\infty} |W_{ij}| \left(\sum_{p=1}^{m} |\Theta^{\alpha}_{i,p}||I(K_{pp};x)| + |\Theta^{\alpha}_{i,0}||I(K_{00};x)| \right) |I(K_{jj};y)| \\
&\leq \sum_{i=m+1}^{\infty}\sum_{j=m+1}^{\infty} |W_{ij}| \left(\sum_{p=1}^{m} |\Theta^{\alpha}_{i,p}| \left(2\sum_{k=1}^{p} \binom{p}{k} |x|^k + 1 \right) + |\Theta^{\alpha}_{i,0}| \right) \left(2\sum_{k=0}^{j} \binom{j}{k} |y|^k + 1 \right).
\end{aligned}$$

If $0 < |x|, |y| \leq 1$, then we get the upper bound for the independence polynomial of a complete bipartite graph as follows

$$|I(K_{ii};x)| = |2\sum_{k=0}^{i} \binom{i}{k} x^k - 1| \leq 2\sum_{k=0}^{i} \binom{i}{k} |x^k| + 1 \leq 2\sum_{k=0}^{i} \binom{i}{k} + 1. \tag{18}$$

□

Theorem 3. Computational Complexity of the proposed algorithm is $O(m^3)$.

Proof. The steps involved in the present algorithm to solve an FPDE can be categorized in three steps:

(1) To compute the operational matrix of fractional integration:
To compute $J_x^{\alpha} I(K_{ii};x)$ in equation (14), let us set that the number of multiplication required is $mA_2 + A_1$. $J^{(\alpha)}$ defined in Theorem 1 is a lower triangular matrix of order $m+1$. The first row is evaluated 1 time, second row is evaluated 2 times and so on. Therefore, total number of computation required to compute $J^{(\alpha)}$ is

$$\frac{(m+1)(m+2)}{2}(mA_2 + A_1).$$

(2) Since the Poisson equation is linear PDE, we solve the system by the Gaussian elimination method. From the mathematical knowledge, the number of operations required in Gaussian elimination for a system of $m+1$ equations is

$$\frac{(m+1)(m+2)(2m+3)}{6}.$$

(3) Finally to compute $w(x,y)$, we need to multiply matrices of order $(1\times m)$, $(m\times m)$ and $(m\times 1)$. Total number of computations required in matrix multiplication is of order $O(m^3)$.

Therefore, the computational complexity of the present algorithm is

$$\frac{(m+1)(m+2)}{2}(mA_2+A_1)+\frac{(m+1)(m+2)(2m+3)}{6}+O(m^3)=O(m^3).$$

□

5. Description of the CBIPC method

A function $w(x,y)\in L^2([0,1]\times[0,1])$ can be approximated by using the independence polynomials of a complete bipartite graph bases as

$$\begin{aligned} w(x,y) &= \sum_{i=0}^{m}\sum_{j=0}^{m} W_{ij} I(K_{ii};x) I(K_{jj};y) \\ &= \Psi^T(x) W \Psi(y). \end{aligned} \tag{19}$$

Here $\Psi(x)$ and $\Psi(y)$ are vectors defined by $\Psi(x)=[K_{00}(x),K_{11}(x),...,K_{mm}(x)]^T$ and $\Psi(y)=[K_{00}(y),K_{11}(y),...,K_{mm}(y)]^T$ and the unknown W is $(m+1)\times(m+1)$ matrix that can be written as

$$W=\begin{pmatrix} W_{00} & W_{01} & ... & W_{0m} \\ W_{10} & W_{11} & ... & W_{1m} \\ . & . & ... & . \\ . & . & ... & . \\ W_{m0} & W_{m1} & ... & W_{mm} \end{pmatrix}.$$

Taking fractional integration of order α and β in equation (19) with respect to x and y, we get

$$\begin{aligned} J_x^{\alpha} w(x,y) &= x^{\alpha}\Psi^T(x)(J^{(\alpha)})^T W\Psi(y), \\ J_y^{\beta} w(x,y) &= y^{\beta}\Psi^T(x) W J^{(\beta)}\Psi(y). \end{aligned}$$

where $J^{(\alpha}$ and $J^{(\beta)}$ are $(m+1)\times(m+1)$ operational matrices of fractional integration of order α and β respectively.

Consider the Poisson equation (2) with boundary condition

$$w(x,0)=f_0(x), w(0,y)=g_0(y), w(x,1)=f_1(x), w(1,y)=g_1(y). \tag{20}$$

Taking fractional integral of order α with respect to x in equation (2), we get

$$w(x,y)-w(0,y)-xD_x w(0,y)+J_x^{\alpha}\left(\frac{\partial^{\beta} w(x,y)}{\partial y^{\beta}}\right)=J_x^{\alpha} g(x,y). \tag{21}$$

Taking fractional integral of order β with respect to y in equation (19), we get

$$J_y^{\beta} w(x,y)-J_y^{\beta} w(0,y)-xJ_y^{\beta} D_x w(0,y)+J_x^{\alpha}(w(x,y)-w(x,0)-yD_y w(x,0))=J_y^{\beta} J_x^{\alpha} g(x,y),$$

Or,

$$J_y^{\beta} w(x,y)+J_x^{\alpha} w(x,y)-J_y^{\beta} w(0,y)-xJ_y^{\beta} D_x w(0,y)-J_x^{\alpha} w(x,0)-yJ_x^{\alpha} D_y w(x,0)=J_x^{\alpha} J_y^{\beta} g(x,y). \tag{22}$$

In equation (22), we substitute the following:

$$\begin{aligned}
&J_y^{\beta} w(x,y) = y^{\beta}\Psi^T(x)WJ^{(\beta)}\Psi(y),\\
&J_x^{\alpha} w(x,y) = x^{\alpha}\Psi^T(x)(J^{(\alpha)})^T W\Psi(y),\\
&J_y^{\beta} w(0,y) = y^{\beta}\Psi^T(0)WJ^{(\beta)}\Psi(y),\\
&J_y^{\beta} D_x w(0,y) = y^{\beta}\Psi^T(0)(D^{(1)})^T WJ^{(\beta)}\Psi(y),\\
&J_x^{\beta} W(x,0) = x^{\alpha}\Psi^T(x)W(J^{(\alpha)})^T\Psi(y),\\
&J_x^{\beta} D_y w(x,0) = x^{\alpha}\Psi^T(x)(J^{(\beta)})^T WD^{(1)}\Psi(y),\\
&J_x^{\alpha} I_y^{\beta} g(x,y) = x^{\alpha}y^{\beta}\Psi^T(x)G\Psi^T(y).
\end{aligned}$$

Here G is a known $(m+1)\times(m+1)$ matrix obtained from (7).
The boundary conditions (20) may be written as:

$$\begin{aligned}
&\Psi^T(x)W\Psi(0) = C_1^T\Psi(x), \quad \Psi^T(0)W\Psi(y) = C_2^T\Psi^T(y),\\
&\Psi^T(x)W\Psi(1) = C_3^T\Psi(x), \qquad \Psi^T(1)W\Psi(y) = C_4^T(x).
\end{aligned} \tag{23}$$

where C_1, C_2, C_3, C_4 are known vectors of dimension $m+1$ and $C_J^{(\alpha)}T = (\int_0^1 f_i(x)\Psi^T(x)dx)Q^{-1}$, and Q is derived from equation (8).
Collocation points are equi-spaced points in the limited domain. If width in between two points are small then we get more accuracy in the solution. These methods helps us while dealing with irregular functions (functions with sudden abrupt changes). In this paper, the collocation points are chosen by the method: $x_i = x_0 + \frac{x_n - x_0}{n+1}i$, in the interval $[a,b] = [x_0, x_m]$. Since the interval in the present paper is $[0,1]$ and it is divided into $m+1$ points, we have chosen the collocation points as $x_i = \frac{i}{n+1}$. Using the collocation points $x_i = \frac{i}{m+1}, y_j = \frac{j}{m+1}, i,j = 0,1,2,...m$, we get $4m$ equations from equation (23) and remaining $(m+1)(m+1) - 4m = (m-1)^2$ equations are obtained from the equation (22).

6. Illustrative examples

This section applies the CBIPC method to obtain a numerical solution of the fractional Poisson equation. To show the efficiency of the present method, we report the absolute error $|w(x_i,y_j) - \tilde{w}(x_i,y_j)|$. The program corresponding to the proposed algorithm is executed using Mathematica 12 in Windows 10 operating system, Processor: Intel(R) Core(TM) i5-4300U CPU @ 1.90GHz 2.49 GHz, RAM: 4.00 GB, 64-bit operating system. CPU timing is computed by executing the code for the proposed algorithm inside the Mathematica function Timing[].

Example 1. We consider the Poisson equation (2) with $\alpha = 4/3$, $\beta = 3/2$ and $g(x,y) = \frac{3x^{2/3}y(y-1)}{\Gamma(\frac{2}{3})} + \frac{4(x-1)x\sqrt{y}}{\sqrt{\pi}}$. Boundary conditions are given as $w(x,0) = w(0,y) = w(x,1) = w(1,y) = 0$. Exact solution is $w(x,y) = xy(x-1)(y-1)$.

Solution 1. : Taking $m = 4$, we approximate the solution of equation (2) by relation (19).
We obtain

$$D^{(1)} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0\\ 2 & 0 & 0 & 0 & 0\\ 2 & 2 & 0 & 0 & 0\\ 3 & 0 & 3 & 0 & 0\\ 4 & 0 & 0 & 4 & 0 \end{pmatrix},$$

$$J^{(4/3)} = \begin{pmatrix} 0.839885 & 0 & 0 & 0 & 0 \\ 0.479934 & 0.359951 & 0 & 0 & 0 \\ 0.335954 & 0.287961 & 0.21597 & 0 & 0 \\ 0.258426 & 0.232584 & 0.199357 & 0.149518 & 0 \\ 0.209971 & 0.19382 & 0.174438 & 0.149518 & 0.112138 \end{pmatrix},$$

$$J^{(3/2)} = \begin{pmatrix} 0.752253 & 0 & 0 & 0 & 0 \\ 0.451352 & 0.300901 & 0 & 0 & 0 \\ 0.322394 & 0.257915 & 0.171943 & 0 & 0 \\ 0.250751 & 0.214929 & 0.171943 & 0.114629 & 0 \\ 0.20516 & 0.182364 & 0.156312 & 0.12505 & 0.0833665 \end{pmatrix}.$$

Collocate equation (22) at $(m-1)^2$ points $x_i = y_j = \frac{i}{m+1}$, $i = j = 1, 2, 3$. Also, we collocate the boundary conditions $\Psi^T(x)W\Psi(0) = 0$ and $\Psi^T(x)W\Psi(1) = 0$ at 2(m-1) points $x_i = y_j = \frac{i}{m+1}$ with $i = j = 0(1)2$ and $\Psi^T(0)W\Psi(y) = 0$ and $\Psi^T(1)W\Psi(y) = 0$ at $2(m+1)$ points $x_i = y_j = \frac{i}{m+1}$ with $i = j = 0(1)4$.
Solving the system of equations, we obtain the solution as

$$\begin{aligned} w(x,y) &= x^4(0.149624 \times 10^{-14}y^4 + 0.186603 \times 10^{-14}y^3 - 0.0166367 \times 10^{-15}y^2 \\ &- 0.00516478 \times 10^{-14}y + 6.444844 \times 10^{-14}) + x^3(0.127752 \times 10^{-14}y^4 - 0.433455 \times 10^{-15}y^3 \\ &+ 0.0519567 \times 10^{-15}y^2 + 0.00616997 \times 10^{-14}y - 9.997558 \times 10^{-14}) + x^2(-0.138865 \times 10^{-14}y^4 \\ &+ 0.39345 \times 10^{-14}y^3 + y^2 - y + 3.84137 \times 10^{-14}) + x(0.116994 \times 10^{-14}y^4 - 0.146598 \times 10^{-14}y^3 \\ &- 0.999877y^2 + 0.999989y - 2.831068 \times 10^{-15}) - 6.59194 \times 10^{-17}y^4 - 2.220446 \times 10^{-16}y^2 \\ &- 3.330669 \times 10^{-16}y - 2.498001 \times 10^{-16} \end{aligned}$$

Similarly, we have computed the solution by taking $m = 6$ and $m = 10$, and presented the absolute error for various values of (x, y) in Table 1 and 2 by taking different collocation points. It is observed that the change of collocation points has the least influence on the accuracy of the solutions obtained using the CBIPC algorithm. In Figure 2, we have presented the absolute error for $m = 4$, $m = 6$ and $m = 10$ and the comparison of approximate solution for $m = 10$ with exact solution.

TABLE 1. Absolute error for Example 1 using collocation points $x_i = \frac{i}{m+1}$, $y_j = \frac{j}{m+1}$.

y	0.2	0.6	1	0.2	0.6	1
x	$m = 6$, CPU time 13.45 sec			$m = 10$, CPU time 19.23 sec		
0	4.776×10^{-16}	1.678×10^{-15}	5.326×10^{-15}	3.786×10^{-13}	3.894×10^{-13}	2.033×10^{-13}
0.2	7.140×10^{-9}	2.331×10^{-8}	4.201×10^{-8}	4.743×10^{-11}	4.742×10^{-11}	4.758×10^{-11}
0.4	1.112×10^{-8}	3.599×10^{-8}	6.445×10^{-8}	9.524×10^{-11}	9.522×10^{-11}	9.531×10^{-11}
0.6	1.191×10^{-8}	3.805×10^{-8}	6.732×10^{-8}	1.431×10^{-10}	1.431×10^{-10}	1.421×10^{-10}
0.8	9.531×10^{-9}	2.948×10^{-8}	5.061×10^{-8}	1.909×10^{-10}	1.908×10^{-10}	1.906×10^{-10}
1.0	3.985×10^{-9}	1.028×10^{-8}	1.434×10^{-8}	2.387×10^{-10}	2.386×10^{-10}	2.382×10^{-10}

TABLE 2. Absolute error for Example 1 using collocation points $x_i = \frac{i}{m^2+1}, y_j = \frac{j}{m^2+1}$.

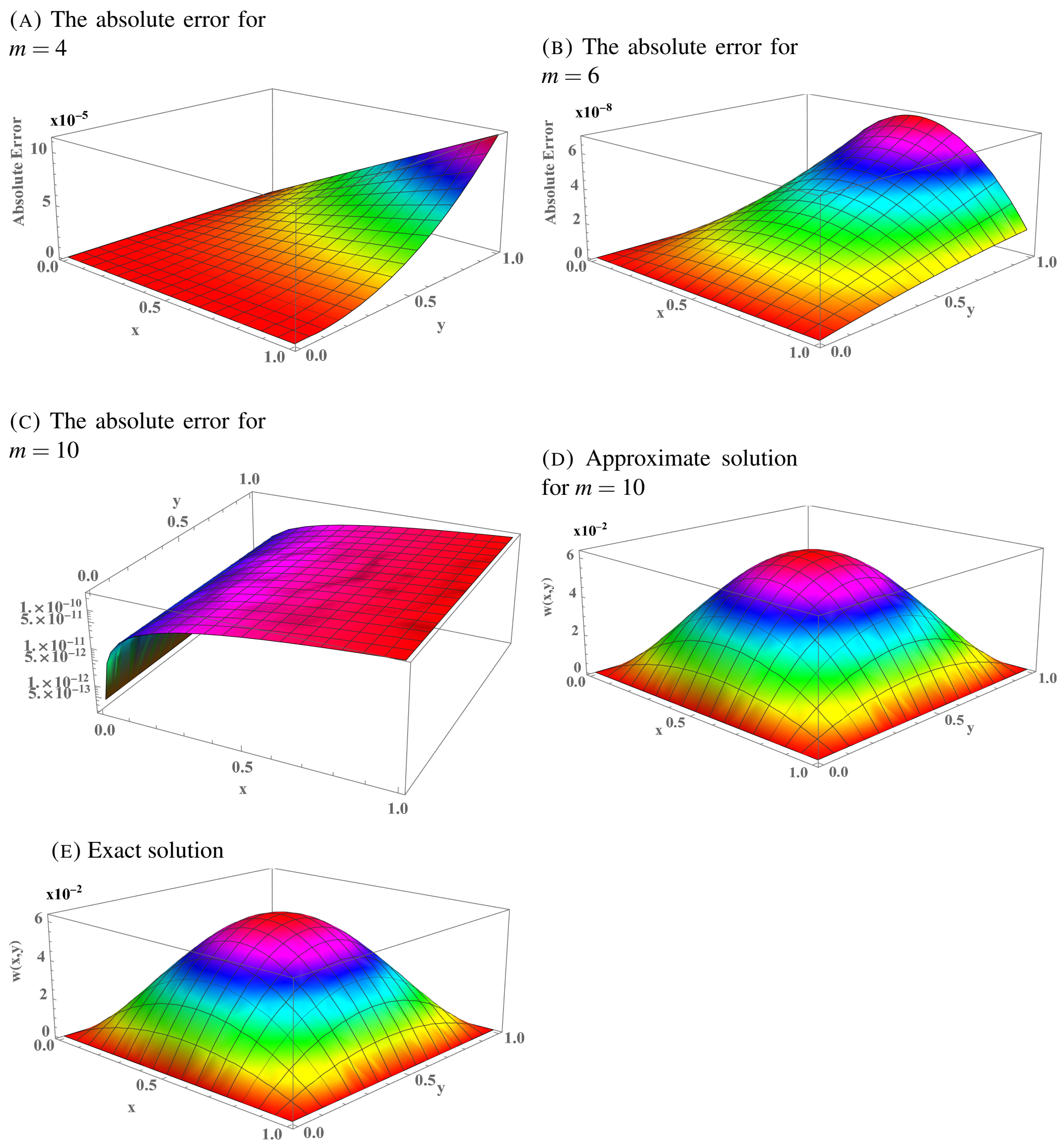
y	0.2	0.6	1	0.2	0.6	1
x	$m = 6$, CPU time 15.22 sec			$m = 10$, CPU time 22.45 sec		
0	5.233×10^{-17}	2.111×10^{-15}	4.452×10^{-15}	3.745×10^{-13}	3.735×10^{-13}	2.121×10^{-13}
0.2	6.782×10^{-9}	2.311×10^{-8}	4.532×10^{-8}	4.643×10^{-11}	4.833×10^{-11}	4.655×10^{-11}
0.4	1.115×10^{-8}	3.521×10^{-8}	6.564×10^{-8}	9.531×10^{-11}	9.511×10^{-11}	9.534×10^{-11}
0.6	1.182×10^{-9}	3.872×10^{-8}	5.431×10^{-8}	1.521×10^{-10}	1.522×10^{-10}	1.331×10^{-10}
0.8	8.112×10^{-9}	2.901×10^{-8}	5.721×10^{-9}	1.302×10^{-10}	1.877×10^{-10}	1.933×10^{-10}
1.0	3.876×10^{-9}	1.811×10^{-9}	1.375×10^{-9}	2.473×10^{-10}	2.346×10^{-10}	2.492×10^{-10}

Example 2. We consider the Poisson equation (2) with $\alpha = 3/2$, $\beta = 4/3$ and $g(x,y) = \frac{8x^{3/2}y^3(8xy-8x-5y+5)}{5\sqrt{\pi}} + \frac{27x^3y^{5/3}(3xy-2x-3y+2)}{10\Gamma\left(\frac{2}{3}\right)}$. Boundary conditions are given as $w(x,0) = w(0,y) = w(x,1) = w(1,y) = 0$. Exact solution is $w(x,y) = x^3y^3(x-1)(y-1)$.
For $m = 4$, $m = 6$ and $m = 10$, the graphs of absolute error are presented in Figure 3. It also projects the comparison of approximate solution for $m = 10$ with exact solution of example 2.

Example 3. We consider the Poisson equation (2) with $\alpha = 2$, $\beta = 3/2$ and $g(x,y) = \frac{4\sqrt{y}(4y-3)\sin(\pi x)\cos(\pi x)}{\sqrt{\pi}} - 4\pi^2 y\left(y-\frac{1}{2}\right)(y-1)\sin(2\pi x)$. Boundary conditions are given as $w(x,0) = w(0,y) = w(x,1) = w(1,y) = 0$. Exact solution is $w(x,y) = \sin(2\pi x)y\left(y-\frac{1}{2}\right)(y-1)$.
In Figure 4(A) and 4(C), the graphs of absolute error for example 3 are presented for $m = 10$ and $m = 14$ respectively. Whereas, Figure 4(B) and 4(D) project approximate solutions for $m = 10$ and $m = 14$ respectively. Figure 4(E) represents the exact solution for Example 3.

7. CONCLUSION

In this paper, a general formulation for the operational matrix of fractional integration has been derived using the independence polynomials of a complete bipartite graph. The fractional derivatives are described in the Caputo sense. The novelty of the present paper is that here we have applied a polynomial generated by a graph to produce an operational matrix of Riemann-Liouville fractional integration. Because the resultant operational matrix is triangular, it reduces the computational complexity of the algorithm. We have also presented the error bound and computational complexity of the projected algorithm. Moreover, while demonstrating the efficiency of the present method with the help of the Poisson equation, we have observed that it could produce a satisfactory amount of accuracy for a smaller number of nodal points. Moreover, even though CPU time consumed depends on the number of collocation points considered by this algorithm, it is observed that CPU time taken is reasonably less. The solutions obtained for the Poisson equation using the present method suggest that this method can be considered as an efficient new addition to numerical analysis literature for solving FDEs.


(A) The absolute error for $m = 4$

(B) The absolute error for $m = 6$

(C) The absolute error for $m = 10$

(D) Approximate solution for $m = 10$

(E) Exact solution

FIGURE 2. Graphs of solution at different values of m for Example 1.

REFERENCES

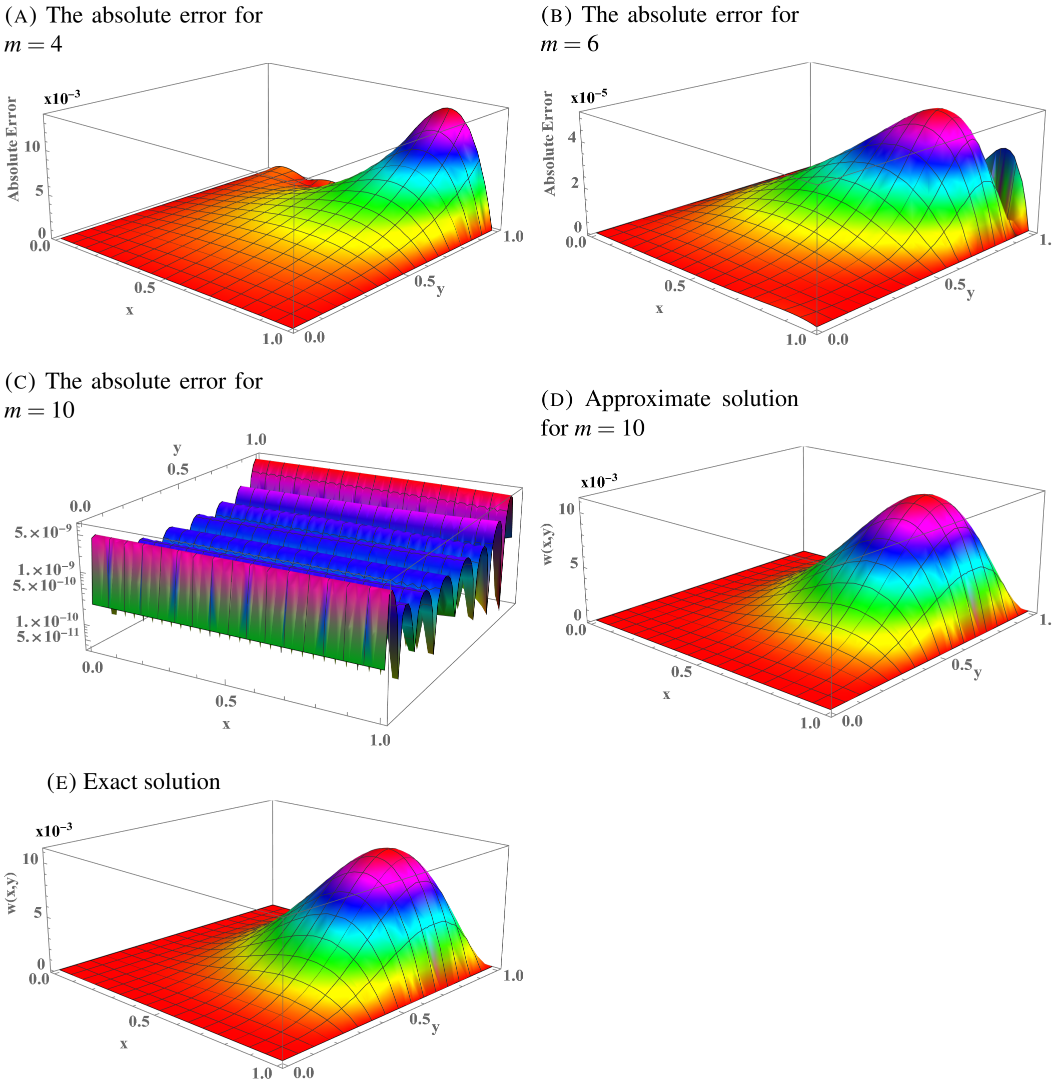
[1] K. Rao, *Introduction to Partial Differential Equations*. Prentice Hall India Learning Private Limited, 2011.

[2] K. Oldham and J. Spanier, *The Fractional Calculus*. Academic Press, New York, 1974.

[3] K. S. Miller and B. Ross, "An introduction to the fractional calculus and fractional differential equations," 1993.

[4] I. Podlubny, *Fractional Differential Equations: An Introduction to Fractional Derivatives*. Elsevier, 1998.

[5] A. A. Kilbas, H. M. Srivastava, and J. Trujillo, *Theory and Applications of Fractional Differential Equations*, vol. 204. Elsevie, North-Holland Mathematics Studies, 2006.


(A) The absolute error for $m = 4$

(B) The absolute error for $m = 6$

(C) The absolute error for $m = 10$

(D) Approximate solution for $m = 10$

(E) Exact solution

FIGURE 3. Graphs of solution at different values of m for Example 2.

[6] D. Kumar, J. Singh, and D. Baleanu, "A new analysis of the fornberg-whitham equation pertaining to a fractional derivative with mittag-leffler type kernel," *The European Physical Journal Plus*, vol. 133, no. 2, 2018.

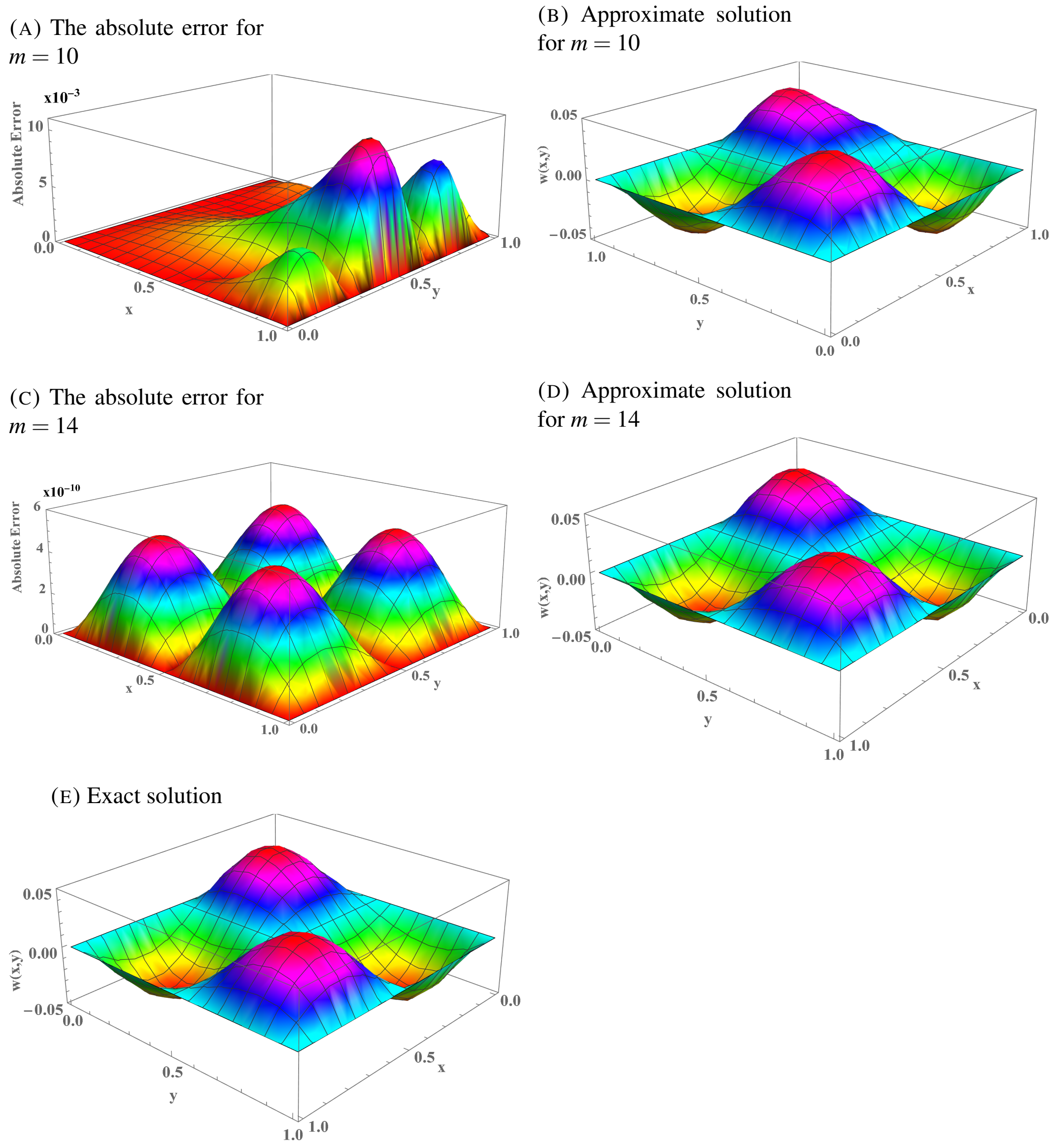
[7] P. Veeresha, D. G. Prakasha, J. Singh, I. Khan, and D. Kumar, "Analytical approach for fractional extended fisher–kolmogorov equation with mittag-leffler kernel," *Advances in Difference Equations*, vol. 2020, no. 1, 2020.

[8] L. Zheng and X. Zhang, *Modeling and Analysis of Modern Fluid Problems*. Academic Press, 2017.

[9] J. He, "Nonlinear oscillation with fractional derivative and its applications," *Conference on Vibrating Engineering'98,Dalian, China*, pp. 288–291, 1998.

[10] J. He, "Some applications of nonlinear fractional differential equations and their approximations," *Bulletin of Science, Technology & Society*, vol. 15, pp. 86–90, 1999.

[11] R. L. Bagley and P. J. Torvik, "A theoretical basis for the application of fractional calculus to viscoelasticity," *Journal of Rheology*, vol. 27, no. 3, pp. 201–210, 1983.


(A) The absolute error for $m = 10$

(B) Approximate solution for $m = 10$

(C) The absolute error for $m = 14$

(D) Approximate solution for $m = 14$

(E) Exact solution

FIGURE 4. Graphs of solution at different values of m for Example 3.

[12] F. Mainardi, “Fractional calculus: Some basic problems in continuum and statistical mechanics,” *Fractals and Fractional Calculus in Continuum Mechanics*, pp. 291–348, 1997.

[13] Y. A. Rossikhin and M. V. Shitikova, “Applications of fractional calculus to dynamic problems of linear and nonlinear hereditary mechanics of solids,” *Applied mechanics review*, vol. 63, no. 1, 2010.

[14] R. Baillie, “Long memory processes and fractional integration in econometrics,” *Journal of Econometrics*, vol. 73, no. 1, pp. 5–59, 1996.

[15] A. Jajarmi, M. Hajipour, and D. Baleanu, “New aspects of the adaptive synchronization and hyperchaos suppression of a financial model,” *Chaos, Solitons and Fractals*, vol. 99, pp. 285–296, 2017.

[16] C. Baishya, S. J. Achar, P. Veeresha, and D. G. Prakasha, "Dynamics of a fractional epidemiological model with disease infection in both the populations," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 31, no. 4, 2021.

[17] C. Baishya, "Dynamics of fractional holling type-ii predator-prey model with prey refuge and additional food to predator," *Journal of Applied Nonlinear Dynamics*, vol. 10, no. 2, pp. 315–328, 2020.

[18] C. Baishya, "Dynamics of fractional stage structured predator prey model with prey refuge," *Indian Journal of Ecology*, vol. 47, pp. 1118–1124, 2020.

[19] R. L. Magin, "Fractional calculus in bioengineering," *Critical Reviews in Biomedical Engineering*, vol. 32, no. 1, pp. 1–104, 2004.

[20] T. Chow, "Fractional dynamics of interfaces between soft-nanoparticles and rough substrates," *Physics Letter A*, vol. 342, pp. 148–155, 2005.

[21] P. K. Gupta and M. Singh, "Homotopy perturbation method for fractional fornberg–whitham equation," *Computers & Mathematics with Applications*, vol. 61, no. 2, pp. 250–254, 2011.

[22] O. Abdulaziz, I. Hashim, and A. Saif, "Series solutions of time-fractional pdes by homotopy analysis method," *International Journal of Differential Euations*, 2008.

[23] R. Darzi and B. Agheli, "Analytical approach to solving fractional partial differential equation by optimal q-homotopy analysis method," *Numerical Analysis and Applications*, vol. 11, no. 2, pp. 134–145, 2018.

[24] S. Kumar, D. Kumar, S. Abbasbandy, and M. M. Rashidi, "Analytical solution of fractional navier–stokes equation by using modified laplace decomposition method," *Ain Shams Engineering Journal*, vol. 5, no. 2, pp. 569–574, 2014.

[25] R. M. Jena and S. Chakraverty, "On the solution of time-fractional coupled system of partial differential equations," *SN Applied Sciences*, vol. 1, no. 12, 2019.

[26] Y. Khan, N. Faraz, A. Yildirim, and Q. Wu, "Fractional variational iteration method for fractional initial-boundary value problems arising in the application of nonlinear science," *Computers & Mathematics with Applications*, vol. 62, no. 5, pp. 2273–2278, 2011.

[27] J. Hristov, "Heat-balance integral to fractional half-time heat diffusion sub-model," *Thermal Science - THERM SCI*, vol. 14, pp. 291–316, 2010.

[28] O. K. K*ü*rkç*ü*, E. Aslan, and M. Sezer, "A novel hybrid method for solving combined functional neutral differential equations with several delays and investigation of convergence rate via residual function," *Computational Methods for Differential Equations*, vol. 7, pp. 396–417, 2019.

[29] O. K. K*ü*rkç*ü*, E. Aslan, and M. Sezer, "An advanced method with convergence analysis for solving space-time fractional partial differential equations with multi delays," *The European Physical Journal Plus*, vol. 134, no. 8, 2019.

[30] E. Aslan, O. K. K*ü*rkç*ü*, and M. Sezer, "A fast numerical method for fractional partial integro-differential equations with spatial-time delays," *Applied Numerical Mathematics*, vol. 161, pp. 525–539, 2021.

[31] F. Liu, V. Anh, and I. Turner, "Numerical solution of the space fractional fokker–planck equation," *Journal of Computational and Applied Mathematics*, vol. 166, no. 1, pp. 209–219, 2004.

[32] V. D. Beibalaev and R. P. Meilanov, "The dirihlet problem for the fractional poisson's equation with caputo derivatives: A finite difference approximation and a numerical solution," *Thermal Science*, vol. 16, no. 2, pp. 385–394, 2012.

[33] O. K. K*ü*rkç*ü*, E. Aslan, and M. Sezer, "An integrated numerical method with error analysis for solving fractional differential equations of quintic nonlinear type arising in applied sciences," *Mathematical Methods in the Applied Sciences*, vol. 42, no. 18, 2019.

[34] W. Chen, L. Ye, and H. Sun, "Fractional diffusion equations by the kansa method," *Computers & Mathematics with Applications*, vol. 59, no. 5, pp. 1614–1620, 2010.

[35] W. Imtiaz, A. Muhammad, and A. Muhammad, "Hybrid b-spline collocation method for solving the generalized burgers-fisher and burgers-huxley equations," *Mathematical problems in engineering*, 2018.

[36] K. Sayevand, A. Yazdani, and F. Arjang, "Cubic b-spline collocation method and its application for anomalous fractional diffusion equations in transport dynamic systems," *Journal of Vibration and Control*, vol. 22, no. 9, pp. 2173–2186, 2016.

[37] H. Sun, W. Chen, C. Li, and Y. Chen, "Finite difference schemes for variable-order time fractional diffusion equation," *International Journal of Bifurcation and Chaos*, vol. 22, 2012.

[38] Z.-J. Fu, W. Chen, and H.-T. Yang, "Boundary particle method for laplace transformed time fractional diffusion equations," *Journal of Computational Physics*, vol. 235, no. C, pp. 52–66, 2013.

[39] H. Khalil and R. Khan, "A new method based on legendre polynomials for solution of system of fractional order partial differential equations," *International Journal of Computer Mathematics*, vol. 91, 2014.

[40] Y. Yang, Y. Ma, and L. Wang, “Legendre polynomials operational matrix method for solving fractional partial differential equations with variable coefficients,” *Mathematical Problems in Engineering*, vol. 2015, pp. 1–9, 2015.

[41] M. Abdelkawy and T. Taha, “An operational matrix of fractional derivatives of laguerre polynomials,” *Walailak Journal of Science and Technology*, vol. 11, pp. 1041–1055, 2014.

[42] E. H. Doha, A. H. Bhrawy, D. Baleanu, and S. S. Ezz-Eldien, “The operational matrix formulation of the jacobi tau approximation for space fractional diffusion equation,” *Advances in Difference Equations*, vol. 2014, no. 1, 2014.

[43] A. Bhrawy and M. Zaky, “A fractional-order jacobi tau method for a class of time-fractional pdes with variable coefficients,” *Mathematical Methods in the Applied Sciences*, vol. 39, no. 7, pp. 1765–1779, 2016.

[44] K. Shah, H. Khalil, and R. A. Khan, “A generalized scheme based on shifted jacobi polynomials for numerical simulation of coupled systems of multi-term fractional-order partial differential equations,” *LMS Journal of Computation and Mathematics*, vol. 20, no. 1, pp. 11–29, 2017.

[45] F. Zhao, Q. Huang, J. Xie, Y. Li, L. Ma, and J. Wang, “Chebyshev polynomials approach for numerically solving system of two-dimensional fractional pdes and convergence analysis,” *Applied Mathematics and Computation*, vol. 313, pp. 321–330, 2017.

[46] W. Han, Y.-M. Chen, D.-Y. Liu, X.-L. Li, and D. Boutat, “Numerical solution for a class of multi-order fractional differential equations with error correction and convergence analysis,” *Advances in Difference Equations*, vol. 2018, 2018.

[47] S. Kumar, P. Pandey, and S. Das, “Numerical solution of two dimensional reaction-diffusion equation using operational matrix method based on genocchi polynomial–part i: Genocchi polynomial and opperatorial matrix,” vol. 20, no. 4, pp. 393–399, 2019.

[48] S. Kumar, P. Pandey, and S. Das, “Operational matrix method for solving nonlinear space-time fractional order reaction-diffusion equation based on genocchi polynomial,” *Reviews in Porous Media: An International Journal*, vol. 11, no. 1, 2020.

[49] K. Parand, S. Hossayni, and J. Rad, “Operation matrix method based on bernstein polynomials for the riccati differential equation and volterra population model,” *Applied Mathematica modelling*, vol. 40, no. 2, pp. 993–1011, 2016.

[50] D. Rostamy, M. Alipour, H. Jafari, and D. Baleanu, “Solving multi-term orders fractional differential equations by operational matrices of bps with convergence analysis,” *Romanian reports in physics*, vol. 65, pp. 334–349, 2013.

[51] J. Hossein and T. Haleh, “Fractional order optimal control problems via the operational matrices of bernstein polynomials,” *UPB Scientific Bulletin, Series A: Applied Mathematics and Physics*, vol. 6, no. 3, pp. 115–128, 2014.

[52] M. Asgari, R. Ezzati, and T. Allahviranloo, “Numerical solution of time-fractional order telegraph equation by bernstein polynomials operational matrices,” *Mathematical Problems in Engineering*, vol. 2016, 2016.

[53] H. S. Ramane, S. C. Shiralashetti, R. A. Mundewadi, and R. B. Jummannaver, “Numerical solution of fredholm integral equations using hosoya polynomial of path graphs,” *American Journal of Numerical Analysis*, vol. 5, no. 1, pp. 11–15, 2018.

[54] B. Agheli, “Approximate solution for solving fractional riccati differential equations via trigonometric basic functions,” *Transactions of A. Razmadze Mathematical Institute*, vol. 172, no. 3, Part A, pp. 299–308, 2018.

[55] O. K. K*ü*rkç*ü*, E. Aslan, and M. Sezer, “On the numerical solution of fractional differential equations with cubic nonlinearity via matching polynomial of complete graph,” *Sādhanā*, vol. 44, 2019.

[56] O. K. K*ü*rkç*ü*, E. Aslan, and M. Sezer, “A novel graph-operational matrix method for solving multidelay fractional differential equations with variable coefficients and a numerical comparative survey of fractional derivative types,” *Turkish Journal of Mathematics*, vol. 43, pp. 373–392, 2019.

[57] S. Kumbinarasaiah, “A new approach for the numerical solution for nonlinear klein–gordon equation,” *SeMA Journal*, 2020.

[58] M. Koc, A.B. nd Cakmak, A. Kurnaz, and K. Uslu, “A new fibonacci type collocation procedure for boundary value problems,” *Advances in difference equation*, vol. 262, 2013.

[59] F. Mirzaee and S. Hoseini, “Application of fibonacci collocation method for solving volterra–fredholm integral equations,” *Applied mathematics and computation*, vol. 273, pp. 637–644, 2016.

[60] W. M. Abd-Elhameed and Y. H. Youssri, “A novel operational matrix of caputo fractional derivatives of fibonacci polynomials: Spectral solutions of fractional differential equations,” *Entropy*, vol. 18, no. 10, 2016.

[61] G. Matthew Ferrin, *Independence Polynomials*. University of South Carolina,Columbia, 2014.

Chandrali Baishya
Department of Studies and Research in Mathematics, Tumkur University, Tumkur, Karnataka, India
Email address: baishyachandrali@gmail.com