Chapter 3

Fibonacci collocation method to solve two-dimensional nonlinear fractional order advection-reaction diffusion equation

3.1 Introduction

In the recent years fractional differential equations have received more attention of the researchers due to its exact description of the physical phenomenon. Many physical phenomena have been described through fractional diffusion equation viz. transport in porous medium, ground water contamination problem through porous medium etc. In this chapter, two-dimensional non-linear advection-reaction diffusion equation (ARDE) is solved with operational matrix method. In the recent years many methods have been developed for this purpose viz., Peaceman and Rachford [101] used alternating direction implicit procedure to solve parabolic and elliptic differential equation in two dimension, Tadjeran and Meerschaert [102] have used ADI approach using a Crank–Nicolson discretization & Richardson extrapolation to find the accurate numerical solution of two-dimensional diffusion equation, Lin et al. [103] have used Preconditioned iterative methods to solve fractional order diffusion equations, Yao [104] has given the exact solution of nonlinear diffusion equation for fluid flow in fractal reservoirs, Wang et al. [105] used $O(Nlog^2N)$ FDM to solve fractional order diffusion equations, Sallam [106] has shown the effect of thermal-diffusion on porous media, Khader [107] has solved the diffusion equation, Molati and

Murakawa [108] have given the exact solution of nonlinear diffusion-convection-reaction equation using a Lie symmetry analysis approach, Jaiswal et al. [109] have given the numerical solution solute transport system.

Transport of solute through porous medium i.e., process of the flow of solute through porous medium have attracted a lot of interests of engineers and scientists including those having expertise on the importance of ground water flow and variety of tertiary oil recovery process. The Ficken diffusive process Matheron and Marsily [110] is generally considered as the transport of solute through porous medium. But in recent years many models have been developed by researchers viz. Mualem [111] had developed porous media model to predict the hydraulic conductivity, Wang and Cheng [112] had developed a multiphase mixture model in capillary porous media, Valencia-López et al. [113] governed mass transfer jump condition at the boundary between a porous medium and a homogeneous fluid, Gómez-Aguilar et al. [114] gave a model on fractional diffusion equation, Chand and Rana [115] have discussed the effect of Brownian diffusion, thermophoresis, and electrophoresis in porous media. Research works on fractional order diffusion equations can be found in the articles of [116, 73], [117, 118] and numerical solution of fractional order PDEs using different numerical techniques are found in ([119, 120, 121, 122, 123]).

In the present chapter, the following space-time fractional order ARDE has been solved

$$
\begin{aligned}
{}_0^C D_t^{\alpha} u(x,y,t) = \vec{\nabla}.\Big[u(x,y,t)\Big\{\hat{i}\big({}_0^C D_x^{\beta-1} u(x,y,t)\big) + \hat{j}\big({}_0^C D_y^{\beta-1} u(x,y,t)\big)\Big\}\Big] \\
- v\{\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y}\} + ku(x,y,t), \;\; 0<\alpha\le 1, 1<\beta\le 2,
\end{aligned}
\tag{3.1}
$$

where $u(x,y,t)$ is the concentration of solute, v is Darcy's velocity, k is the coefficient of the reaction term. For $\alpha = 1$ and $\beta = 2$, the solute transport model (3.1) becomes the integer order classical model. The advection-dispersion equation generally governs the solute transport in soil, reservoir and aquifer. In the considered model, a nonlinear dispersion term with reaction term and flow of the solute in two dimension is taken. The considered problem can be physically represented as, consider the aquifer is of finite length, and pollutants enter to the ground water through porous media from point source. As the concentration of the pollutants is higher than the concentration of the ground water, the diffusion will take place. The pollutant will move vertically downwards to the bottom of the ground water and will spread horizontally. In the considered model, same horizontal and lateral flow of the ground water and dispersion coefficient function of solute concentration have been taken in both longitudinal and horizontal directions. The considered model with boundary conditions like Drichlet, Neumann and Cauchy boundary conditions can easily be solved with the numerical method. In this chapter the model (3.1) has been solved with following initial and boundary conditions as,

$$\begin{aligned}
u(x,y,0) &= 0,\ \ 0 \le x \le 1, 0 \le y \le 1, \\
u(0,y,t) &= a_O,\ \ 0 \le y \le 1, 0 < t \le 1, \\
\frac{\partial u(1,y,t)}{\partial x} &= 0,\ \ 0 \le y \le 1, 0 < t \le 1, \\
\frac{\partial u(x,0,t)}{\partial y} &= 0,\ \ 0 \le x \le 1, 0 < t \le 1, \\
\frac{\partial u(x,1,t)}{\partial y} &= 0,\ \ 0 \le x \le 1, 0 < t \le 1.
\end{aligned} \tag{3.2}$$

From the above given conditions, it can be said that initially the ground water is pollute free, at any time t on $x = 0$ there is certain amount of pollutants present in water and from rest of the conditions it can be said that tangents at different points and in different directions are zero.

In this chapter, the two-dimensional nonlinear fractional order ARDE is solved using spectral collocation method. Generally there are three versions of spectral method viz. collocation, tau and Galerkin methods. In the spectral method the solution is approximated through series of polynomials as $\sum a_{ijk}\phi_i\phi_j\phi_k$, where ϕ is set of polynomials and coefficients a_{ijk} are to be obtained with suitable spectral method. In the collocation method, after expanding residual, initial and boundary conditions as series of functions we collocate it at certain collocation points. Here, the solution is approximated by using Fibonacci polynomial as basis function, as the solution consists one time and two spatial directions, so the Kronecker products of basis vector are used to approximate the solution. Generally researchers prefer operational matrix of orthogonal polynomial but in recent years many methods have been developed using Fibonacci polynomials due to its less computational time and higher accuracy. Koç et al. [83] used Fibonacci operational matrix to solve integer order boundary value problem and the same authors Koç et al. [84] have generalized the pantograph equations with the help of Fibonacci polynomial, Mirzaee and Hoseini [124] used the operational matrix to solve Fredholm–Volterra integral equations in two-dimensional spaces, Elhameed and Youssri [85] have used the Fibonacci operational matrix to solve fractional order ordinary differential equation. In the last decade operational matrices have been developed by many researchers to solve the differential equations viz., Razzaghi and Yousefi [125] use the Legendre wavelets operational matrix for integration, Babolian and Fattahzadeh [126] used the Chebyshev wavelet operational matrix to solve differential equation. Saadatmandi and Dehghan [86] developed operational matrix with shifted Legendre polynomials to solve ordinary differential equation. As the problem consists of three variables, so there is the possibility that number of constants may be large during numerical computation as compared to one dimensional problem, which may increase the computational time during use of orthogonal polynomials. But number of zeros in the Fibonacci matrix is greater than any other orthogonal operational matrix, which causes the increase of accuracy and reduces the computational time. Due to this advantage, such operation matrix has been suggested during the solution of two-dimensional nonlinear space-time fractional order diffusion equation.

This chapter is arranged as follows. In section 3.2 some required definitions and notations have been given. Section 3.3 consist of derivatives in terms of Fibonacci operational matrix. In section 3.4, the proposed method has been expressed to solve arbitrary diffusion equation in fractional order. In section 3.5, the method has been applied on three existing problems and compare the analytical solutions with the numerical solution obtained from the proposed method to show the accuracy of the method. In section 3.6, the method has been applied on the considered model (3.1) and the various results are discussed through graphical plots. The overall work is illustrated through the section Conclusion.

3.2 Preliminaries

3.2.1 Three-dimensional Fibonacci polynomial

Three dimensional form of Fibonacci polynomial is defined as

$$\phi_{lmn}(t,x,y) = F_l(t)F_m(x)F_n(y), \quad l,m,n = 0,1,2... \ . \tag{3.3}$$

Here in above equation $F_l(t)$, $F_m(x)$ and $F_n(y)$ are well-known Fibonacci polynomials of order l, m and n.

3.2.2 Kronecker product

The Kronecker product denoted by '$\otimes$' was first introduced by German mathematician Leopold Kronecker, which is an operation on two matrices of arbitrary size resulting in a block matrix. If A and B represent linear transformations $V_1 \to W_1$ and $V_2 \to W_2$, respectively, then $A \otimes B$ represents the tensor product of the two maps, $V_1 \otimes V_2 \to W_1 \otimes W_2$. The applications of Kronecker products can be found in signal processing, image processing, semi definite programming, and quantum computing. Kronecker products are going to be a very effective way to look at fast linear transforms ([127, 128, 129, 130, 131]).

Remark 1:
Let $A_{n\times m}$ and $B_{p\times q}$ be two $n\times m$ and $p\times q$ matrices, respectively. The Kronecker product of A and B is denoted by $A\otimes B = kron(A,B)$ and is defined by

$$A\otimes B = \begin{pmatrix} a_{11}B & a_{12}B\cdots & \cdots & a_{1m}B \\ a_{21}B & a_{22}B\cdots & \cdots & a_{2m}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}B & a_{n2}B & \cdots & a_{nm}B \end{pmatrix},$$

where $A\otimes B$ is an $(n\times p)\times(m\times q)$ matrix with

$$A = \begin{pmatrix} a_{11} & a_{12}\cdots & \cdots & a_{1n+1} \\ a_{21} & a_{22}\cdots & \cdots & a_{1n+1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{n+1n+1} \end{pmatrix}_{(n\times m)}, \quad B = \begin{pmatrix} b_{11} & b_{12}\cdots & \cdots & b_{1m} \\ b_{21} & b_{22}\cdots & \cdots & b_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nm} \end{pmatrix}_{(p\times q)}.$$

Remark 2:
The matrices A_1, A_2 and B_1, B_2 with appropriate dimensions satisfy the following important property

$$(A_1A_2)\otimes(B_1B_2) = (A_1\otimes B_1)(A_2\otimes B_2). \tag{3.4}$$

3.2.3 Function approximation

Consider $f(x,y,t)$ be any arbitrary function such that $f(x,y,t)\in C([0,1]\times[0,1]\times[0,1])$, then $f(x,y,t)$ can be approximated in the combination of the Fibonacci polynomial as

$$f(x,y,t) \cong \sum_{i}^{n+1}\sum_{j}^{n+1}\sum_{k}^{n+1} C_{ijk}F_i(t)F_j(x)F_k(y) = \phi_n^T(t)C\psi_n(x,y), \tag{3.5}$$

where

$$\phi_n(t) = [F_1(t), F_2(t), F_3(t)...F_{n+1}(t)]^T \text{ and } \psi_n(x,y) = \phi_n(x)\otimes\phi_n(y),$$

$$\psi_n(x,y) = [F_1(x)F_1(y), F_1(x)F_2(y)...F_1(x)F_{n+1}(y), F_2(x)F_1(y)...F_{n+1}(x)F_{n+1}(y)],$$

and $C_{(n+1)\times(n+1)^2}$ be an $(n+1)\times(n+1)^2$ dimensional constant matrix defined by

$$C = \begin{pmatrix} c_{11} & c_{12}\cdots & \cdots & c_{1(n+1)^2} \\ c_{21} & c_{22}\cdots & \cdots & c_{2(n+1)^2} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n+11} & c_{n+12} & \cdots & c_{n+1(n+1)^2} \end{pmatrix}_{(n+1)\times(n+1)^2}.$$

3.3 Derivative in terms of Fibonacci operational matrices

The integer order derivative of $\phi_n(x)$ is [85]

$$\frac{d\phi_n(x)}{dx} = M^1\phi_n(x), \tag{3.6}$$

where $M^1 = (m_{ij}^1)$ is the Fibonacci operational matrix of derivative of order $(n+1)\times(n+1)$, with

$$m_{ij}^1 = \begin{cases} (-1)^{\frac{i-j+3}{3}} j, & if\ i \geq j, (i+j) odd, \\ 0, & otherwise. \end{cases}$$

In order to obtain the differentiation of $\phi_n(x)$ of an arbitrary integer, the equation (3.6) can be re-written as

$$\frac{d^k\phi_n(x)}{d^kx} = M^k\phi_n(x) = (M^1)^k\phi_n(x), \tag{3.7}$$

where k is any arbitrary integer.

Now to obtain the derivative of $\phi_n(x)$ in any arbitrary fractional order, we have [85]

$${}_0^CD^\alpha\phi_n(x) = x^{-\alpha}M^\alpha\phi_n(x), \tag{3.8}$$

where Fibonacci operational matrix of α order derivative $M^\alpha = (m_{ij}^\alpha)$ is defined by

$$M^{\alpha} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \xi_{\alpha}(\lceil\alpha\rceil,1) & \xi_{\alpha}(\lceil\alpha\rceil,\lceil\alpha\rceil) & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \xi_{\alpha}(i,1) & \cdots & \xi(i,i) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \xi_{\alpha}(n+1,1) & \xi_{\alpha}(n+1,2) & \xi_{\alpha}(n+1,3) & \cdots & \xi_{\alpha}(n+1,n+1) \end{pmatrix}_{(n+1)\times(n+1)},$$

where

$$m_{ij}^{\alpha} = \begin{cases} \xi_{\alpha}(i,j), & i \geq \lceil\alpha\rceil, i \geq j, \\ 0, & otherwise, \end{cases}$$

with

$$\xi_{\alpha}(i,j) = \sum_{k=\lceil\alpha\rceil}^{i} \frac{(-1)^{\frac{k-j+1}{2}} k!(\frac{i+k-1}{2})!}{(\frac{i-k-1}{2})!(\frac{k-j+1}{2})!(\frac{k+j+1}{2})!\Gamma(k+1-\alpha)}.$$

The α order derivative of $\psi(x,y)$ w.r.to x is

$$\begin{aligned} {}_0^C D_x^{\alpha} \psi_n(x,y) &= {}_0^C D_x^{\alpha} \big(\phi_n(x) \otimes \phi_n(y)\big) \\ &= {}_0^C D_x^{\alpha} \phi_n(x) \otimes \phi_n(y) \\ &= (x^{-\alpha} M^{\alpha} \phi_n(x)) \otimes (I\phi_n(y)) \\ &= x^{-\alpha}(M^{\alpha} \otimes I)(\phi_n(x) \otimes \phi_n(y)) \\ &= x^{-\alpha} M_x^{\alpha}(\phi_n(x) \otimes \phi_n(y)) \\ &= x^{-\alpha} M_x^{\alpha} \psi_n(x,y), \end{aligned} \tag{3.9}$$

where $I_{n+1\times n+1}$ is an identity matrix and $M_x^{\alpha} = (M^{\alpha} \otimes I)$.

Similarly the α order derivative of $\psi_n(x,y)$ w.r.to y is

$$\begin{aligned} {}_0^C D_y^{\alpha} \psi_n(x,y) &= {}_0^C D_y^{\alpha} \big(\phi_n(x) \otimes \phi_n(y)\big) \\ &= \phi_n(x) \otimes {}_0^C D_y^{\alpha} \phi_n(x,y) \\ &= (I\phi_n(x)) \otimes (y^{-\alpha} M^{\alpha} \phi_n(y)) \\ &= y^{-\alpha}(I \otimes M^{\alpha})(\phi_n(x) \otimes \phi_n(y)) \\ &= y^{-\alpha} M_y^{\alpha} \phi_n(x) \otimes \phi_n(y) \\ &= y^{-\alpha} M_y^{\alpha} \psi_n(x,y), \end{aligned} \tag{3.10}$$

where $M_y^\alpha = (I \otimes M^\alpha)$.

3.4 Method of solution of 2D fractional order diffusion equation

Consider nonlinear fractional order two-dimensional ARDE equation as

$$ {}_0^C D_t^\alpha u(x,y,t) = \omega\Big({}_0^C D_x^\beta u(x,y,t), {}_0^C D_y^\gamma u(x,y,t), \frac{du(x,y,t)}{dx}, \frac{du(x,y,t)}{dy}, u(x,y,t), a(x,y)\Big), \tag{3.11}$$

where $0 \le \alpha \le 1$ and $1 \le \beta, \gamma \le 2$,

with initial condition as

$$u(x,y,0) = g_1(x,y), \tag{3.12}$$

and the boundary conditions as

$$\begin{aligned} u(0,y,t) &= g_2(y,t),\\ u(1,y,t) &= g_3(y,t),\\ u(x,0,t) &= g_4(x,t),\\ u(x,1,t) &= g_5(x,t). \end{aligned} \tag{3.13}$$

Let us approximate $u(x,y,t)$ in the combination of Fibonacci polynomial as

$$u(x,y,t) \cong \phi_n^T(t) C \psi_n(x,y),$$

then from the equations (3.6), (3.8), (3.9) and (3.10) the fractional order derivatives of $u(x,y,t)$ with respect to different variables are

$$\begin{aligned} {}_0^C D_t^\alpha u(x,y,t) &\cong {}_0^C D_t^\alpha \big(\phi_n^T(t) C \psi_n(x,y)\big)\\ &= \big({}_0^C D_t^\alpha \phi_n(t)\big)^T C \psi_n(x,y)\\ &= (t^{-\alpha} M^\alpha \phi_n(t))^T C \psi_n(x,y)\\ &= t^{-\alpha} \phi_n^T(t) (M^\alpha)^T C \psi_n(x,y), \end{aligned} \tag{3.14}$$

and

$$\begin{aligned} {}_0^C D_x^{\beta} u(x,y,t) &\cong {}_0^C D_x^{\beta}\big(\phi_n^T(t) C \psi_n(x,y)\big) \\ &= \phi_n^T(t) C \big({}_0^C D_x^{\beta} \psi_n(x,y)\big) \\ &= x^{-\beta} \phi_n(t) C M_x^{\beta} \psi_n(x,y), \end{aligned} \tag{3.15}$$

$$\begin{aligned} {}_0^C D_y^{\gamma} u(x,y,t) &\cong {}_0^C D_y^{\gamma}\big(\phi_n^T(t) C \psi_n(x,y)\big) \\ &= \phi_n^T(t) C \big({}_0^C D_y^{\gamma} \psi_n(x,y)\big) \\ &= y^{-\gamma} \phi_n(t) C M_y^{\gamma} \psi_n(x,y). \end{aligned} \tag{3.16}$$

Hence with the help of equations (3.14), (3.15) and (3.16), we have the following Residual

$$\begin{aligned} R(x,y,t) = t^{-\alpha} \phi_n^T(t) (M^{\alpha})^T C \psi_n(x,y) - \\ \omega\{x^{-\beta} \phi_n(t) C M_x^{\beta} \psi_n(x,y), y^{-\gamma} \phi_n(t) C M_y^{\gamma} \psi_n(x,y), \phi_n(t) C M_x^{1} \psi_n(x,y) \\ , \phi_n(t) C M_y^{1} \psi_n(x,y)\}. \end{aligned} \tag{3.17}$$

Again from the initial and boundary conditions we obtain

$$\begin{aligned} \phi_n^T(0) C \psi_n(x,y) &= g_1(x,y), \\ \phi_n^T(t) C \psi_n(0,y) &= g_2(y,t), \\ \phi_n^T(t) C \psi_n(1,y) &= g_3(y,t), \\ \phi_n^T(t) C \psi_n(x,0) &= g_4(x,t), \\ \phi_n^T(t) C \psi_n(x,1) &= g_5(x,t). \end{aligned} \tag{3.18}$$

Now to obtain the unknown matrix $C_{(n+1)\times(n+1)^2}$, let us apply the spectral collocation method so that the residue must vanishes certain collocation points, which can be chosen as $x_i = y_i = t_i = \frac{2i-1}{2n+1}$. Let us collocate the residual at $(n-1)\times(n-1)\times n$ points, so that equation (3.17) gives

$$R(x_i, y_j, t_k) = 0, \quad 1 \le i \le n-1, \;\; 1 \le j \le n-1 \;\; 1 \le k \le n. \tag{3.19}$$

Hence we have $n(n-1)^2$ algebraic equations, and rest of this system is obtained from initial and boundary conditions as

$$
\begin{aligned}
\phi_n^T(0)C\psi_n(x_i, y_i) &= g_1(x_i, y_i), \quad 1 \le i \le n+1, \ 1 \le j \le n+1, \\
\phi_n^T(t_k)C\psi_n(0, y_j) &= g_2(y_j, t_k), \quad 1 \le j \le n, \ 1 \le k \le n, \\
\phi_n^T(t_k)C\psi_n(1, y_j) &= g_3(y_j, t_k), \quad 1 \le j \le n, \ 1 \le k \le n, \\
\phi_n^T(t_k)C\psi_n(x_i, 0) &= g_4(x_i, t_k), \quad 1 \le i \le n, \ 1 \le k \le n, \\
\phi_n^T(t_k)C\psi_n(x_i, 1) &= g_5(x_i, t_k), \quad 1 \le i \le n, \ 1 \le k \le n.
\end{aligned} \tag{3.20}
$$

Hence from equations (3.19) and (3.20), we have set of $(n+1)^3$ equations which can be solved with the help of Newton's method.

3.5 Error analysis and accuracy of the method

In this section, the concerned numerical method will be validated by applying it on the following three existing two-dimensional fractional order PDEs and compare the obtained results with their analytical results through absolute error.

The absolute error is defined as

$$E_R(x, y, t) = |U_{exact}(x, y, t) - u(x, y, t)|.$$

The maximum error in x for fixed y and t is defined by

$$MaxE_R(x, 0.5, 1) = \max_{0 \le x \le 1} |E_r(x, 0.5, 1)|.$$

Similarly maximum error in y for a fixed x and t is defined by

$$MaxE_R(0.5, y, 1) = \max_{0 \le y \le 1} |E_r(0.5, y, 1)|.$$

The numerical order of convergence is calculated by the following formula

$$\text{Order} = \frac{log(\frac{MaxE_R(N_1)}{MaxE_R(N_2)})}{log(\frac{N_2}{N_1})},$$

where $MaxE_R(N)$ is maximum absolute error for N degree approximation.

3.5.1 Numerical Examples

Example 1.

Consider the fractional order PDE [102]

$$\frac{\partial u(x,y,t)}{\partial t} = f_1(x,y)\,{}_0^C D_x^{1.8} u(x,y,t) + f_2(x,y)\,{}_0^C D_y^{1.6} u(x,y,t) + f_3(x,y,t), \tag{3.21}$$

where $x \in [0,1]$, $y \in [0,1]$, $t \in [0,1]$, $f_1(x,y) = \frac{\Gamma(2.2)x^{2.8}y}{6}$, $f_2(x,y) = \frac{2xy^{2.6}}{\Gamma(4.6)}$ and $f_3(x,y,t) = -(1+2xy)e^{-t}x^3y^{3.6}$, under the initial condition

$$u(x,y,0) = x^3y^{3.6}, \tag{3.22}$$

and the boundary conditions

$$\begin{aligned} u(0,y,t) &= 0,\\ u(1,y,t) &= e^{-t}y^{3.6},\\ u(x,0,t) &= 0,\\ u(x,1,t) &= e^{-t}x^3, \end{aligned} \tag{3.23}$$

Table 3.1: Maximum absolute error at $t = 1$.

	n	Maximum absolute error (0≤y≤1)	order	CPU time (Sec)
	4	2.26463e-3	-	1.46
x=0.5	5	1.98837e-4	10.9019	3.35
	6	7.43523e-6	18.0245	9.59

Table 3.2: Maximum absolute error at $t = 1$.

	n	Maximum absolute error (0≤x≤1)	order	CPU time (Sec)
	4	3.00923e-4	-	1.46
y=0.5	5	6.37084e-5	6.95758	3.35
	6	1.75276e-6	19.7076	9.59

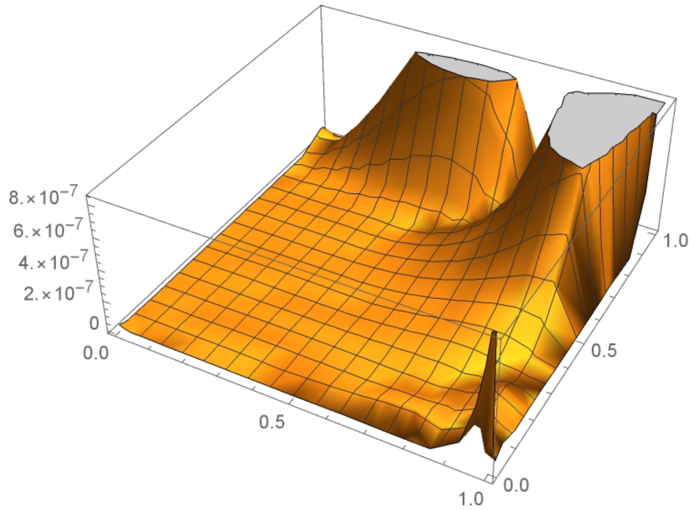

FIGURE 3.1: The absolute error of example.1 at time $t = 0.5$ and $n = 10$.

which has the exact solution is $u(x, y, t) = e^{-t}x^3y^{3.6}$. The error analysis has been exhibited through Table 3.1 and Table 3.2 for $x = 0.5$, $0 \leq y \leq 1$ and $y = 0.5$, $0 \leq x \leq 1$, respectively for the order of approximation $n = 4, 5, 6$ at $t = 1$. The tables clearly confirm that the order of convergence of the proposed method increases as the degree of Fibonacci polynomial in x and y increases. Again absolute error decreases with the increase of the order of approximation of the polynomial n, which clearly shows the effectiveness of our concerned numerical method. Again the data of CPU time shows that it takes minimum time to obtain accurate result which shows the method is computationally effective to solve two dimensional fractional order PDE. Fig. 3.1 shows that the numerical solution is approximately accurate to the exact solution even for smaller value of $n = 10$. The better convergence can be achieved with increasing n. Thus comparing the numerical solution with the exact solution, it can be said that the proposed method is very much efficient than other methods. It also shows that developed numerical method is much more accurate than the finite difference method given by [102]. It can be seen that in the cases before and after extrapolation, the maximum error is 10^{-4} and 10^{-5}, respectively, which is higher than maximum error obtained by our proposed method at small value of n. Hence the proposed method is of higher accuracy.

Example 2.

The fractional order PDE

$$\frac{\partial u(x,y,t)}{\partial t} = h_1(x,y)\,{}_0^C D_x^{1.9} u(x,y,t) + h_2(x,y)\,{}_0^C D_y^{1.6} u(x,y,t) + h_3(x,y,t), \tag{3.24}$$

where $x \in [0,1]$, $y \in [0,1]$, $t \in [0,1]$, $h_1(x,y) = \frac{x^3y^{1.4}}{\Gamma(3.9)}$, $h_2(x,y) = \frac{x^{1.1}y^3}{\Gamma(3.6)}$, $h_3(x,y,t) = -(1+2x61.1y^{1.4})e^{-t}x^{2.9}y^{2.6}$, with the initial condition

$$u(x,y,0) = x^{2.9}y^{2.6} \tag{3.25}$$

and boundary conditions

$$\begin{aligned} u(0,y,t) &= 0, \\ u(1,y,t) &= e^{-t}y^{2.6}, \\ u(x,0,t) &= 0, \\ u(x,1,t) &= e^{-t}x^{2.9}, \end{aligned} \tag{3.26}$$

Table 3.3: Maximum absolute error at $t = 1$.

	n	Maximum absolute error (0≤y≤1)	order	CPU time (Sec)
	4	1.02861e-3	-	1.297
x=0.5	5	3.83727e-4	4.41882	3.61
	6	4.29519e-5	12.0109	9.72

Table 3.4: Maximum absolute error at $t = 1$.

	n	Maximum absolute error (0≤x≤1)	order	CPU time (Sec)
	4	3.116991e-4	-	1.297
y=0.5	5	1.96882e-4	2.05891	3.61
	6	1.91298e-5	12.0109	9.72

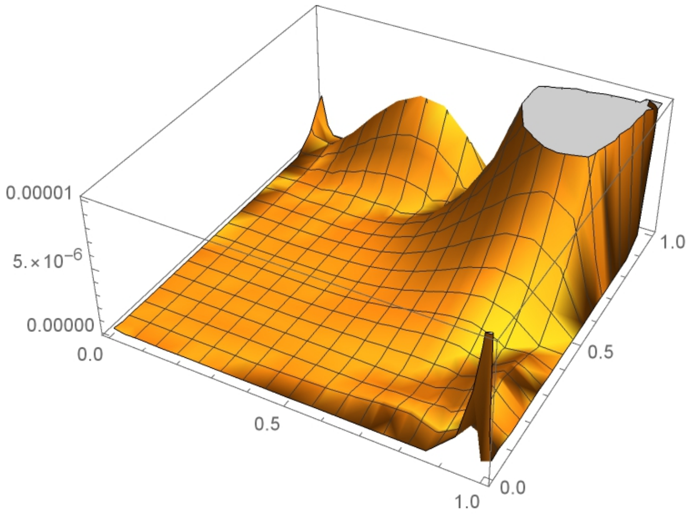

FIGURE 3.2: The absolute error of example.2 at time $t = 0.5$ and $n = 10$.

has the exact solution $u(x, y, t) = e^{-t}x^{2.9}y^{2.6}$ [132]. The errors, order of convergence and the CPU time required for solving the PDE with concerned method are given through Table 3.3 and Table 3.4 for $x = 0.5$, $0 \le y \le 1$ and $y = 0.5$, $0 \le x \le 1$, respectively for $n = 4, 5, 6$ at $t = 1$. The tables clearly exhibit that our concerned method is computationally effective in less CPU time. From Fig 3.2, it is said that the accuracy of the numerical solution of the problem (3.24) is much higher even for small order of approximation when the proposed method is applied. Proposed method provides more accurate results as compared to previously solved results [132] where non-standard finite difference method and standard finite difference method have been used to find the maximum error as 10^{-1} and 10^{-3}, respectively.

Example 3. Consider the following two dimensional form of the Fisher diffusion equation as

$$\frac{\partial u(x, y, t)}{\partial t} = \nabla^2 u(x, y, t) + u(1 - u), \tag{3.27}$$

where ∇^2 is two-dimensional Laplacian operator. The equation (3.27) with initial condition

$$u(x, y, 0) = [1 + exp((x - y/\sqrt{2})/\sqrt{6})]^{-2}, \tag{3.28}$$

and boundary conditions

$$
\begin{aligned}
u(0,y,t) &= [1+exp(\frac{(\frac{-y}{\sqrt{2}})-(\frac{5}{\sqrt{6}}t)}{\sqrt{6}})]^{-2},\\
u(1,y,t) &= [1+exp(\frac{(1-\frac{y}{\sqrt{2}})-(\frac{5}{\sqrt{6}}t)}{\sqrt{6}})]^{-2},\\
u(x,0,t) &= [1+exp(\frac{x-(\frac{5}{\sqrt{6}}t)}{\sqrt{6}})]^{-2},\\
u(x,1,t) &= [1+exp(\frac{(x-\frac{1}{\sqrt{2}})-(\frac{5}{\sqrt{6}}t)}{\sqrt{6}})]^{-2},
\end{aligned}
\tag{3.29}
$$

has an exact solution $u(x,y,t) = [1+exp(\frac{(x-\frac{y}{\sqrt{2}})-(\frac{5}{\sqrt{6}})t}{\sqrt{6}})]^{-2}$ [133], where $(x,y,t) \in R^2 \times R^+$.

TABLE 3.5: Maximum absolute error at $t = 1$.

	n	Maximum absolute error (0≤y≤1)	order	CPU time (Sec)
	4	0.000896455	-	10.06
x=0.5	5	0.000496986	2.64353	58.48
	6	0.000371642	1.59405	160.82

TABLE 3.6: Maximum absolute error at $t = 1$.

	n	Maximum absolute error (0≤x≤1)	order	CPU time (Sec)
	4	0.000879324	-	10.06
y=0.5	5	0.000526421	3.24333	58.48
	6	0.000376182	1.84305	160.82

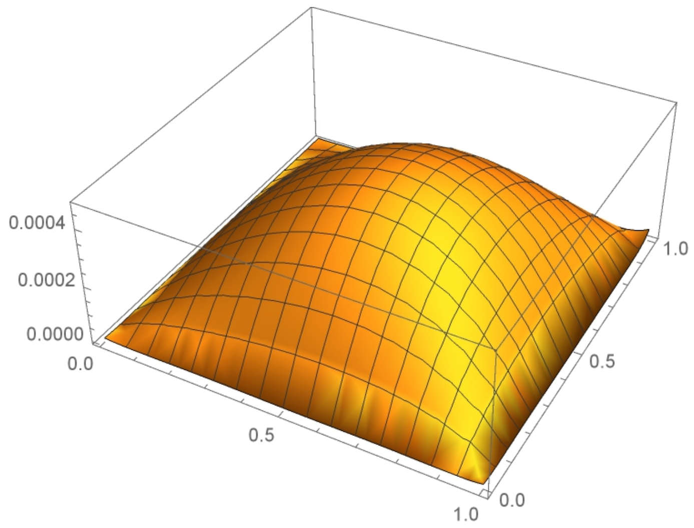

FIGURE 3.3: The absolute error of Example 3 at time $t = 0.5$ and $n = 10$.

Tables 3.5 and 3.6 clearly describe that the concerned method is even computationally efficient during the solution of the two-dimensional nonlinear fractional order PDE. It is seen from Tables 3.5 and 3.6 that the maximum absolute errors decrease as the order of the polynomial increases, but the decay rate is very slow as compared to the first two problems. Therefore the order of convergence decreases for both the cases. This actually happens due to the presence of non-linearity term in the reaction term. The Fig 3.3 shows that the absolute error between exact solution and approximate solution using the proposed method at $t = 0.5$ and $n = 10$. On increasing the value of n the error can be reduced and thus the accurate result of the nonlinear problem 3 is obtained.

3.6 Solution of the two dimensional space-time ARDE

After being a validation of the accuracy and effectiveness of the method, the author has motivated to solve the proposed two-dimensional nonlinear space time fractional order model (3.1) with the initial boundary conditions (3.2) to find the solute concentration in the fluid. The residual of the equation (3.1) can be obtained as

$$
\begin{aligned}
R(x,y,t) = {} & t^{-\alpha}\phi_n^T(t)(M^\alpha)^T C\psi_n(x,y) \\
& - \phi_n^T(t)C\psi_n(x,y)[x^{-\beta}\phi_n^T(t)CM_x^\beta\psi_n(x,y) + y^{-\beta}\phi_n^T(t)CM_y^\beta\psi_n(x,y)] \\
& - [x^{\beta-1}\phi_n^T(t)CM_x^1\psi(x,y)\phi_n^T(t)CM_x^{\beta-1}\psi_n(x,y) \\
& + y^{\beta-1}\phi_n^T(t)CM_y^1\psi_n(x,y)\phi_n^T(t)CM_y^{\beta-1}\psi_n(x,y)] \\
& + v[\phi_n^T(t)CM_x^1\psi_n(x,y) + \phi_n^T(t)CM_y^1\psi_n(x,y)] - k\phi_n^T(t)C\psi_n(x,y).
\end{aligned}
\tag{3.30}
$$

The initial and boundary conditions (3.2) are reduced to

$$
\begin{aligned}
&\phi_n^T(0)C\psi_n(x_i,y_i) = 0, \quad 1 \le i \le n+1, \ 1 \le j \le n+1, \\
&\phi_n^T(t_k)C\psi_n(0,y_j) = a_o, \quad 1 \le j \le n, \ 1 \le k \le n, \\
&\phi_n^T(t_k)CM_x^1\psi_n(1,y_j) = 0, \quad 1 \le j \le n, \ 1 \le k \le n, \\
&\phi_n^T(t_k)CM_y^1\psi_n(x_i,0) = 0, \quad 1 \le i \le n, \ 1 \le k \le n, \\
&\phi_n^T(t_k)CM_y^1\psi_n(x_i,1) = 0, \quad 1 \le i \le n, \ 1 \le k \le n.
\end{aligned}
\tag{3.31}
$$

Now running i, j from 1 to $n-1$ and k from 1 to n for the residual given in equation (3.30) by taking collocation points discussed in section 4 together with equations (3.31), we get system of $(n+1)\times(n+1)^2$ algebraic equations, which can be solved for $n=10$ with the help of Newton's method using Mathematica software.

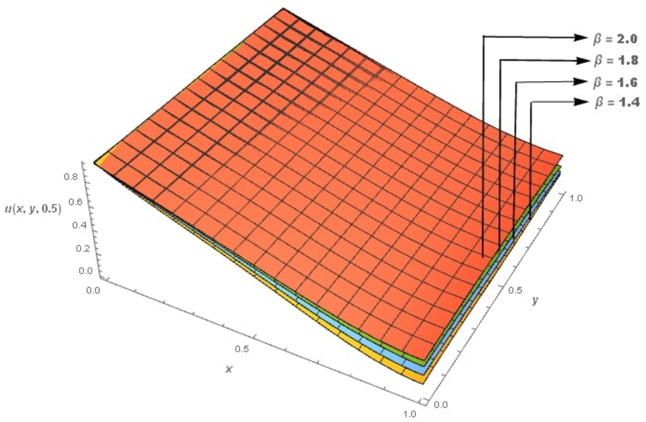

FIGURE 3.4: Solute concentration for non-conservative system with sink term when $\alpha = 1$, $v = 0$ at $t = 0.5$.

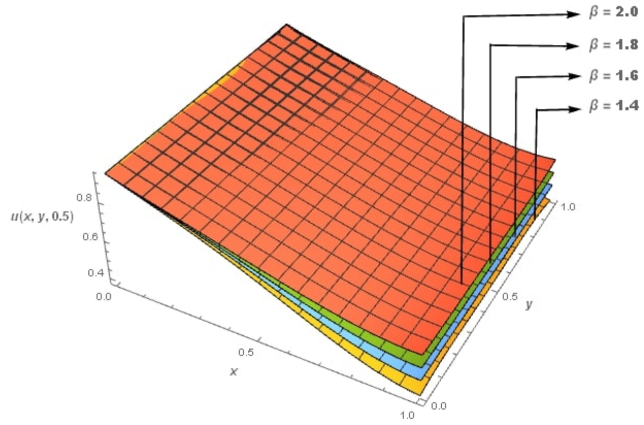

FIGURE 3.5: Solute concentration for non-conservative system with sink term for $\alpha = 1$, $v = 1$ at $t = 0.5$.

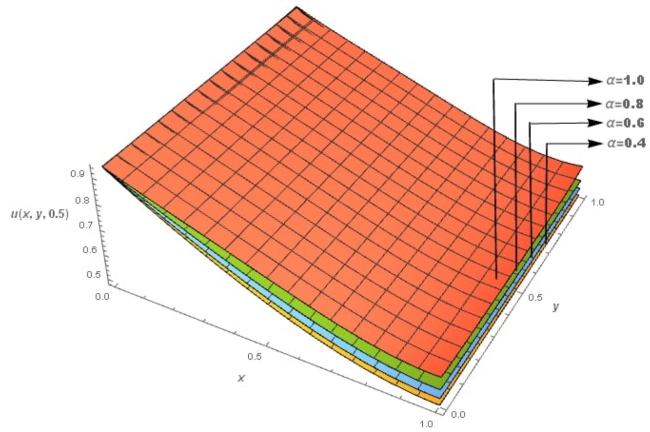

FIGURE 3.6: Pollute concentration for non-conservative system with sink term for $\beta = 2$ $v = 1$ at $t = 0.5$.

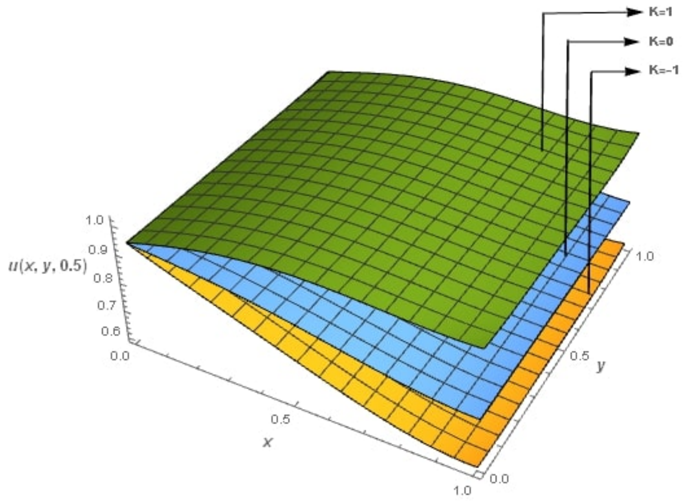

FIGURE 3.7: Pollute concentration for conservative and non-conservative systems for $\alpha = 1$ $\beta = 2$ at $t = 0.5$.

3.6.1 Results and Discussion

In this section, the numerical values of the solute concentration vs. x and y at time $t = 0.5$ are calculated numerically for various values of α and β for conservative and non-conservative systems in presence/absence of advection terms. The effect of reaction and advection terms on the pollute concentration in ground water during presence and absence of advection terms are displaced through Fig. 3.4-3.7. The effects of advection term on pollute profile for two-dimensional space fractional order ARDE (α=1) are shown through Fig. 3.4 and Fig. 3.5 for various values of β. It is seen that as β decreases the pollute concentration in ground water decreases with and without the presence of advection term. Again decrease of magnitude of pollute concentration will be increased due to the presence of advection term ($v = 1$) which is physically relevant. As the spatial order goes from integer to fractional order, pollute will be diffused more due to the effect of advection term.

A comparison of the variations of pollute concentration vs. x and y for space fractional ARDE ($\alpha = 1$) and time fractional ARDE ($\beta = 2$) for two-dimensional problem is done through Fig. 3.5 and Fig. 3.6, respectively. It is seen in both the occasions the concentration decreases as the system goes to fractional order from the standard order. The only difference is that the decay rate of concentration is higher for time fractional model as compared to the space fractional model, therefore diffusion rate of pollute will be less for time fractional ARDE as compared to space fractional ARDE. The variations of concentration for standard order two dimensional ARDE (α=1, β=2) are shown in Fig. 3.7 for various values of x and y for conservative and non-conservative systems. It is clear from the figure that the pollute concentration will be much higher and lower as compared to conservative system ($k = 0$) due to the presence of source ($k = 1$) and sink ($k = -1$) respectively. Thus the rate of diffusion of the pollute will be more due to the effect sink term.

3.7 Conclusion

In this chapter spectral collocation method is used to solve the two-dimensional space-time fractional order ARDE with non-linear diffusive term using Fibonacci operational matrix. The effects on pollute concentration in ground water for different fractional order spatial derivative and time derivative have been shown graphically on concentration during the presence and absence of advection term for both conservative and non-conservative systems. The efficiency and effectiveness of the proposed method are validated by comparing the numerical solutions obtained by the proposed method with existing analytical solutions of two fractional order linear diffusion equations and another two-dimensional non-linear Fisher equation through error analysis. The salient feature of the study is the tabular presentation, order of convergence and minimum CPU time required for the numerical computations. The important feature of the present study on the concerned mathematical model is the graphical exhibition of the higher rate of pollute diffusion for spatial fractional case as compared to time fractional case and also due to effect of advection and sink terms.