Chapter 2

Numerical solution of nonlinear space-time fractional order advection-reaction-diffusion equation

2.1 Introduction

In last few years, the fractional order differential equation (FDE) has attracted its attention by the researchers for its enormous applications in many areas of science, engineering and economics. There are many physical phenomena like contamination in ground-water, transportation in porous media which are discussed through fractional calculus. Another reason behind its popularity among the researchers is its non-local behavior and the resulting memory effect on the system. This has given a mileage to the fractional order system over the standard order system for which engineers apply it to various real life problems. Fractional differential equation provides the exact description of a nonlinear phenomenon. Due to the wide and important applications of FDEs, many researchers have developed numerical algorithms to solve such types of equations. Hence, several analytical and numerical methods are available in the literature such as domain decomposition method [69], Wavelet operational method [71], homotopy analysis method [72], variational iterative method [73], etc. Moreover, there are several numerical methods to solve different types of fractional diffusion equations [74, 75, 76, 77, 78, 73, 79, 80, 81, 82], etc.

The Fibonacci polynomial is an extension of Fibonacci numbers. The Fibonacci polynomials can easily be generated by recurrence relation. Fibonacci numbers are known for a long time, but the Fibonacci polynomials become important in the world of polynomials very recently. Many methods have been developed by using the Fibonacci polynomials viz., method used for solving ordinary BVP by Koç et al. [83] in the year 2013, a matrix method used to solve generalized pantograph equation by the same authors [84] in the year 2014, Fibonacci operational method is used to solve FDE by Abd-Elhameed et al. [85] in the year 2016. In recent years, many operational matrices have been developed to solve fractional order partial differential equation viz., Legendre operational matrix [86], Bessel's operational matrix [87], Chebyshev operational matrix [88], etc. The operational matrix of Fibonacci polynomials gives much accurate result than orthogonal polynomials [83]. The orthogonal polynomial takes a long time during computation as compared to the Fibonacci polynomial. A large number of zeros in the operational matrix increases the efficiency of the solution and makes calculation easier. In the proposed method, the integer order power of variable is approximated keeping fractional power separately which gives the accurate result of the derivative. While solving fractional order partial differential equations (FPDEs), Fibonacci polynomial gives more accurate result than a corresponding solution using orthogonal polynomial even on taking the small degree of the Fibonacci polynomial. Therefore, from the literature survey, it is clear that the method is of scientific interest over the other methods during the solutions of FPDEs.

It is known to us that water is one of the primary needs of the living body. Although two/third of the earth's surface is water, in which 97.5% is being salted water and 2.5% is being fresh water. Only 0.3% is the liquid form of the fresh water on the surface. Surface and ground waters both are getting polluted due to many reasons. The contamination of the water bodies is called water pollution. The surface water contamination is caused after the discharge of wastewater into the surface water. The groundwater contamination is normally happens when the man-made products viz. gasoline, oil, road salt and chemical are constantly diffused through the porous medium into the groundwater. Pesticides and fertilizers are the examples which are diffused into groundwater. Road salt, a toxic substance from mining site, may also seep into the groundwater.

The advection reaction diffusion equation (ARDE) is widely used in science and engineering as a mathematical model for computational purposes such as in oil reservoir simulation, global weather prediction, transport of mass and energy, chemical transformation, etc. The solute spreads within the fluid by molecular diffusion. The random collision of the solute molecule with fluid molecule causes the diffusion and produces a flux from the higher concentrated areas to the lower concentrated areas. The advective term describes the bulk movement of solute particles in the direction of fluid flow at the rate equal to fluid velocity. In addition to advective transport, solute spreads within the fluid in the porous medium by molecular diffusion. Bear and Bachmat [89] stated that the coefficient of molecular diffusion in an isotropic medium is dependent on the diffusion coefficient of the particular solute in water and the tortuosity of the medium. The rate of molecular diffusion is independent of groundwater velocity and diffusion occurs even in the absence of fluid movement.

Many models and methods for solving groundwater contamination problems have been developed. In 2005, Younes [90] developed ELLAM method with moving grid to solve nonlinear ARDE in one dimension to show that contamination transfer of biological, chemical, radioactive processes. Analytical solution of advection diffusion equation with variable coefficient which describes the transport of solute through the porous medium was given by Ahmed et al. [91]. In 2009, Guerrero et al. [92] have developed a method with the help of classic integral transform technique to find the analytic solution of MLCT subject to sequential fractional order equation in the finite media.

In this chapter, an endeavor is made to solve the following nonlinear space-time fractional order ARDE:

$$\begin{gathered} {}_0^C D_t^{\alpha} u(x,t) = u(x,t) {}_0^C D_x^{\beta} u(x,t) - v\frac{\partial u(x,t)}{\partial x} + ku(x,t), \\ 0 \le x \le 1,\ 0 < \alpha \le 1\ ,\ 1 < \beta \le 2, \end{gathered} \tag{2.1}$$

with initial and boundary conditions as

$$u(x,0) = \Psi_1(x),\ \ 0 \le x \le 1, \tag{2.2}$$

$$u(0,t) = \Psi_2(t),\ \ t > 0, \tag{2.3}$$

$$u(1,t) = \Psi_3(t)\ \ t > 0, \tag{2.4}$$

where $u(x,t)$ is concentration of the solute in fluid at finite distance x and at time t, the parameters α and β represent time and space fractional order derivatives respectively, v represents the uniform velocity of fluid in x-direction, k represents the coefficient of source/sink term for the production or loss of solute within system, and $\Psi_1(x)$, $\Psi_2(t)$, $\Psi_3(t)$ are known functions which represent the distribution of solute concentration initially and concentration on the boundary points of the medium at any time t. If $\alpha = 1$ and $\beta = 2$, the frictional order ARDE is reduced to the classical ARDE with nonlinear diffusion term. The presence of the nonlinear diffusive term increases the solute concentration in comparison to the linear diffusion equation in the fluid. The positive power of $u(x, t)$ in the diffusive term causes the slow diffusion in comparison to general linear diffusion equation and will increase the solute concentration in the fluid. The physical phenomenon like fast diffusion or slow diffusion is very much relevant for the porous media and thus the nonlinear term in the diffusivity plays an important role from the physical point of view as compared to a linear model. This has motivated me to solve a porous media problem of nonlinear order in fractional order system.

In the above problem, the Dirichlet condition is considered during the solution of equation (2.1), even though it can be solved with the well posed and ill posed Neumann and Cauchy boundary conditions. Deng et al. [93] discussed about well posedness of boundary value fractional diffusion model with various types of boundary conditions. It is found from the literature that to solve the two-dimensional problem numerically, three versions of the spectral methods are found viz., collocation, Galerkin and Tau methods. In the spectral method, the solution is expressed as the series of polynomials like $\sum a_{ij}\phi_i\phi_j$, where ϕ is a set of the polynomials and the coefficients are obtained with the suitable spectral method. In collocation method, the residue corresponding to partial differential equations (PDEs) have to be zero at certain collocation points. In Tau method, the boundary conditions are applied after expanding residual function in the form of series of polynomials. In Galerkin method, we choose the basis function which satisfy initial and boundary conditions and after that the residual is made orthogonal with the considered basis functions [94, 95, 96, 97]. In the present endeavor, an attempt has been taken to solve the space-time fractional order ARDE equation (1) using Fibonacci collocation method with its operational matrices. The effect of reaction term on the solution profile for different parametric values of α and β due to presence and absence of advection term is depicted through figures for different particular cases.

The present chapter is arranged as fallows. Some preliminary definitions are given in section 2.2. Section 2.3 consists of some information related to integer and fractional order derivatives with Fibonacci operational matrix. Numerical method to solve FPDEs has been discussed in the section 2.4. In the section 2.5, the discussed method is validated by applying it on three existing examples. Finally the discussed method is used to solve the considered model (2.1) and behavior of diffusion profile has been discussed in the section 2.6. Overall work is concluded in the section 2.7.

2.2 Preliminaries

As it has been discussed in the chapter 1 that any square integrable function can be approximated as series of Fibonacci polynomial, an approximation is considered as

$$f(x) \simeq \sum_{k=1}^{n+1} c_k F_k = C^T \phi_n(x), \tag{2.5}$$

where $C^T = [c_1, c_2, c_3......c_{n+1}]$ and $\phi_n(x) = [F_1, F_2, F_3....F_{n+1}]^T$, which can be extended to the function of two variables. For a function $u(x,t) \in L^2([0,1]\times[0,1])$, the approximation using Fibonacci polynomial can be obtained as

$$u(x,t) \cong \sum_{i=i}^{n+1}\sum_{j=1}^{n+1} C_{ij} F_i(x) F_j(t) = \phi_n^T(x) C \phi_n(t),$$

where

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

2.3 Generalization of Fibonacci Operational Matrix to fractional order derivative

The derivative of $\phi_n(x)$ can be written as [85]

$$\frac{d\phi_n(x)}{dx} = M^1 \phi_n(x), \tag{2.6}$$

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

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

Taking $n = 6$, the operational matrix M^1 becomes

$$M^1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ -1 & 0 & 3 & 0 & 0 & 0 \\ 0 & -2 & 0 & 4 & 0 & 0 \\ 1 & 0 & -3 & 0 & 5 & 0 \end{pmatrix},$$

which is the operational matrix for integer order derivatives. The main aim is to generalize the operational matrix of the integer order to the fractional order. The equation (2.6) can be re-written as

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

where k is any positive integer.

The fractional order derivative of the Fibonacci polynomial vector $\phi_n(x)$ is written in the form of Fibonacci operational matrix for any $\alpha > 0$ as [85]

$$^C_0D^\alpha_x\phi_n(x) = x^{-\alpha}M^\alpha\phi_n(x), \tag{2.7}$$

where $M^\alpha = (m^\alpha_{ij})$ is Fibonacci operational matrix of derivatives of order α, which is defined as

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

where

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

with

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

For instant, if $\alpha = 0.5$ and $n = 4$, we have

$$M^{0.5} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1.12838 & 0 & 0 & 0 \\ -0.501502 & 0 & 1.50451 & 0 & 0 \\ 0 & -0.150451 & 0 & 1.80541 & 0 \\ -0.169794 & 0 & -0.109614 & 0 & 2.06332 \end{pmatrix}$$

To validate the effectiveness of the chosen Fibonacci operational matrix for fractional order, a comparison is shown with the existing Caputo fractional derivatives for t^5 given in Table 2.1. It is clearly shown from the table that the results are identical. This concludes that the fractional order derivative can be approximated by the Fibonacci operational matrix of order $(n+1) \times (n+1)$.

TABLE 2.1: Absolute error of t^5 for different fractional order derivatives.

α	0.5	1.5	2.5	3.5	4.5
For x=0.2	$8.45678\ 10^{-16}$	$9.09689\ 10^{-15}$	$5.85532\ 10^{-13}$	$1.61648\ 10^{-12}$	$3.19886\ 10^{-11}$
For x=0.6	$5.55112\ 10^{-16}$	$1.42109\ 10^{-14}$	$7.81597\ 10^{-14}$	$9.9476\ 10^{-14}$	$4.26326\ 10^{-14}$
For x=1.0	$8.88178\ 10^{-16}$	$1.42109\ 10^{-14}$	$3.55271\ 10^{-14}$	$9.9476\ 10^{-14}$	$5.68434\ 10^{-14}$

2.4 Numerical method to solve the fractional order partial differentiation equation

Consider the nonlinear FPDE with variable coefficient as

$${}_0^C D_t^{\alpha} u(x,t) = \omega\big({}_0^C D_x^{\beta} u(x,t), u(x,t), a(x)\big), \quad x \in (0,1), \alpha \in (q_1, q_1+1], \quad \beta \in (q_2, q_2+1], \tag{2.8}$$

with initial conditions as

$$u^{(i)}(x,0) = a_i, \quad i = 0,1,2...,q_1,$$

and boundary conditions as

$$\begin{cases} u^{(j)}(0,t) = b_j, u^{(j)}(1,t) = c_j, \\ \qquad\qquad j = 0,1,2,...,\frac{q_2-1}{2}, \quad \text{if } q_2 \text{ is odd}, \\ u^{(j)}(0,t) = d_j, u^{(j)}(1,t) = e_j, u^{(\frac{q_2}{2})}(0,t) = f_j, \\ \qquad\qquad j = 0,1,2...,\frac{q_2}{2}-1, \quad \text{if } q_2 \text{ is even}. \end{cases}$$

Let us approximate $u(x,t)$ by Fibonacci polynomial as

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

Then the fractional order derivative of $u(x,t)$ with the help of equation (2.7) is

$$\begin{aligned} {}_0^C D_t^{\alpha} u(x,t) &\cong {}_0^C D_t^{\alpha}\big(\phi_n^T(x)C\phi_n(t)\big) \\ &= \phi_n^T(x)C{}_0^C D_t^{\alpha}\phi_n(t) \\ &= t^{(-\alpha)}\phi_n^T(x)CM^{\alpha}\phi_n(t) \end{aligned} \tag{2.9}$$

and

$$\begin{aligned} {}_0^C D_x^{\beta} u(x,t) &\cong {}_0^C D_x^{\beta}\big(\phi_n^T(x)C\phi_n(t)\big) \\ &= \big({}_0^C D_x^{\beta}\phi_n(x)\big)^T C\phi_n(t) \\ &= (x^{-\beta}M^{\beta}\phi_n(x))^T C\phi_n(t) \\ &= x^{-\beta}\phi_n^T(x)(M^{\beta})^T C\phi_n(t). \end{aligned} \tag{2.10}$$

Hence with the help of equations (2.9) and (2.10), the Residual term of given nonlinear FPDE is obtained as

$$\begin{aligned} R(x,t) = {} & t^{(-\alpha)}\phi_n^T(x)CM^{-\alpha}\phi_n(t) \\ & -\omega(x^{-\beta}\phi_n^T M^{\beta}C\phi_n(t), \phi_n^T(x)C\phi_n(t), a(x)). \end{aligned}$$

From the above prescribed initial conditions, we get

$$\phi_n^T(x)((M^1)^i)^T C\phi_n(0) = a_i \;\; i = 1,2,3...,q_1,$$

and from the boundary conditions, we get

$$\begin{aligned} &\phi_n^T(0)C(M^1)^j\phi_n(t) = b_j, \\ &\phi_n^T(1)C(M^1)^j\phi_n(t) = c_j, \quad j = 0,1,2...,\frac{q_2-1}{2}, \end{aligned}$$

for the case when q_2 is odd and

$$\begin{aligned} &\phi_n^T(0)C(M^1)^j\phi_n(t) = d_j, \\ &\phi_n^T(1)C(M^1)^j\phi_n(t) = e_j, \\ &\phi_n(0)CM^{(\frac{q_2}{2})}\phi_n(t) = f_j, \;\; j = 0,1,2...,\frac{q_2}{2}-1, \end{aligned}$$

for the case when q_2 is even.

To apply the Spectral collection method, it is required that $R(x,t)$ must vanishes at certain

collocation points
$x = \frac{i}{n+1}, i = 1, 2, 3..., (n - q_2)$, and $t = \frac{j}{n+1}, j = 1, 2, 3..., (n - q_1)$.
Hence

$$R(\frac{i}{n+1}, \frac{j}{n+1}) = 0, \quad i = 1, 2, 3..., (n - q_2), \quad j = 1, 2, 3..., (n - q_1) \tag{2.11}$$

and from boundary conditions either one can compare the coefficients or collocate the initial and boundary conditions at the chosen collocation points including corner points for boundary conditions. Hence from equation (2.11) and boundary conditions, one may get $(n + 1) \times (n + 1)$ equations with $(n + 1) \times (n + 1)$ unknowns, which have been solved by using Newton's method.

2.5 Error Analysis

In this section, the proposed method is applied to three existing problems to validate the efficiency, effectiveness and simplicity of the proposed method by calculating the maximum absolute error and order of convergence of the proposed method by formulae given below before applying it to the concerned space-time fractional order ARDE (2.1).

$$\text{Maximum Absolute Error } E_n(t) = \underset{0 \le x \le 1}{Max} |U(x,t) - u_n(x,t)|,$$

$$\text{and order of convergence} = \frac{log(\frac{E_{n_1}(t)}{E_{n_2}(t)})}{log(\frac{n_2}{n_2})},$$

where $U(x, t)$ be the exact and $u_n(x, t)$ is approximate solutions for n degree of approximation at particular time t. The CPU time is also calculated for the purpose stated above.

Example 1. Let us consider following PDE

$$\frac{\partial^{\frac{1}{2}} u(x,t)}{\partial x^{\frac{1}{2}}} + \frac{\partial^{\frac{3}{2}} u(x,t)}{\partial t^{\frac{3}{2}}} = \frac{4\sqrt{t}x + 2\sqrt{x}t^2}{\sqrt{\pi}},$$

with the initial and boundary conditions $u(x, 0) = 0$, $u(x, 1) = x$, $u(0, t) = 0$, $u(1, t) = t^2$ whose exact solution [98] is given by $u(x, t) = xt^2$. Now using equations (2.9) and (2.10) and solving for $n = 2$, we get the residual term as

$$R(x,t) = \frac{2}{(3\sqrt{\pi})}\{(6c_{13}\sqrt{t} + \sqrt{x}(3c_{21} + 3c_{22}t - 3t^2 + 3c_{23}(1 + t^2 + 2\sqrt{t}\sqrt{x}) - 6\sqrt{t}\sqrt{x} + 4c_{31}x + 4c_{32}tx) + c_{33}(4x^{3/2} + 4t^2x^{3/2} + 6\sqrt{t}(1 + x^2))\}.$$

The boundary conditions give

$$c_{11} + c_{13} + c_{21}x + c_{23}x + c_{31}(1 + x^2) + c_{33}(1 + x^2) = 0,$$

$$c_{11} + c_{12} - x + c_{21}x + c_{22}x + c_{31}(1 + x^2) + c_{32}(1 + x^2) + 2(c_{13} + c_{23}x + c_{33}(1 + x^2)) = x,$$

$$c_{11} + c_{31} + (c_{12} + c_{32})t + (c_{13} + c_{33})(1 + t^2) = 0,$$

$$c_{11} + c_{21} + 2c_{31} + (c_{12} + c_{22} + 2c_{32})t - t^2 + (c_{13} + c_{23} + 2c_{33})(1 + t^2) = t^2.$$

Finding the coefficients from above four equations and choosing $x = \frac{1}{3}$ and $t = \frac{1}{3}$ as collation points for Residual, we get nine linear algebraic equations with nine unknowns. On solving these equations, we get

$$u(x,t) = -2.9697910^{-17} + t^2(-8.5209110^{-17} + x) \approx xt^2,$$

which implies that the approximate and exact solutions are similar.

Example 2. Consider the following Burger's equation

$$\frac{\partial u(x,t)}{\partial t} + u(x,t)\frac{\partial u(x,t)}{\partial x} = \frac{\partial^2 u(x,t)}{\partial x^2},$$

with initial condition as

$$u(x,0) = 2x,\ 0 \leq x \leq 1,$$

and boundary conditions as

$$u(0,t) = 0,\ \ \frac{\partial u(0,t)}{\partial x} = \frac{2}{1+2t},\ \ t \geq 0,$$

whose exact solution is $u(x,t) = \frac{2x}{1+2t}$ [99].

Table 2.2 is drawn to show the maximum absolute error, order of convergence and time taken to solve the Example 2 with the proposed method at different value of degree of approximation for $t = 0.5$.

TABLE 2.2: Maximum absolute error at $t = 0.5$ for Example 2.

n	Maximum Absolute error	Order of Convergence	CPU time (sec)
3	0.014314	-	0.438
5	8.41549e-4	5.54	0.609
7	2.88222e-5	10.02	1.484

From above Table 2.2, one can easily say that for proposed method has higher order convergence rate with very less computational time even for solving nonlinear problems and also it is very much visible that on increasing the degree of approximation n, the absolute error can further be reduced.

Example 3. Consider a non-linear PDE

$$\frac{\partial u(x,t)}{\partial t} = (1+u)\frac{\partial^2 u(x,t)}{\partial x^2} - \frac{\partial u(x,t)}{\partial x} - u^2 + u,$$
$$0 < x \leq 0, \quad t \geq 0,$$

with the initial condition

$$u(x,0) = e^x,$$

and the boundary conditions

$$u(0,t) = e^t, u(1,t) = e^{t+1}.$$

The exact solution of this problem is $e^x E_1(t)$ [100], where $E_1(t)$ is Mittag–Leffler function. Table 2.3 expresses the maximum absolute error, order of convergence and computational time taken by the proposed method on applying it to Example 3.

TABLE 2.3: Maximum absolute error at $t = 0.5$ for example 3.

n	Maximum Absolute error	Order of Convergence	CPU time (sec)
3	0.00844524	-	0.374
5	6.0358e-5	9.62	0.484
7	1.92246e-7	17.08	1.062

It is clearly found from the table that the accuracy of our proposed method is much higher as compared to the method used in [100] while solving the same problem. The maximum absolute error is 10^{-3} in [100] whereas in our case it is 10^{-3} for $n = 3$ and as the degree of approximation increases it is rapidly decreases in less CPU time.

2.6 Solution of the space-time fractional order ARDE

After validation of the efficiency, effectiveness and accuracy of the proposed numerical method in the previous section, it is used to get the solution of space-time fractional order ARDE (2.1) under the initial condition

$$u(x,0) = x(1-x),$$

and boundary conditions

$$u(0,t) = 0, \;\; u(1,t) = 0.$$

Now to solve the ARDE (2.1) using the proposed numerical method let us approximate $u(x,t)$ by Fibonacci polynomial as

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

where

$$C = \begin{pmatrix} c_{11} & c_{12} \cdots & \cdots & c_{1n+1} \\ c_{21} & c_{22} \cdots & \cdots & c_{1n+1} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{n+1n+1} \end{pmatrix}_{(n+1)\times(n+1)}$$

Using equations (2.9) and (2.10), the residual term for the equation (2.1) will be

$$\begin{aligned} R(x,t) =& t^{(-\alpha)}\phi_n^T(x)CM^{\alpha}\phi_n(t) \\ & - x^{-\beta}(\phi_n^T(x)C\phi_n(t))(\phi_n^T(x)(M^{\beta})^T C\phi_n(t)) \\ & + v(\phi_n^T(x)(M^1)^T C\phi_n(t)) - k(\phi_n^T(x)C\phi_n(t)). \end{aligned} \tag{2.12}$$

Using the prescribed initial and boundary conditions, we get

$$\phi_n^T(x)C\phi_n(0) = x(1-x), \tag{2.13}$$

$$\phi_n^T(0)C\phi_n(t) = 0, \tag{2.14}$$

$$\phi_n^T(0)C\phi_n(t) = 0. \tag{2.15}$$

Let us collocate the residual term at the collocation points $x = \frac{k}{n+1}$, $k = 1, 2, 3...(n-1)$ and $t = \frac{h}{n+1}$,$h = 1, 2, 3...n$. In addition to these points, $x = 1$ and $t = 1$ are to be included to collocate the boundary conditions. Finally we get the system of $(n+1)^2$ numbers of algebraic equations with $(n+1)^2$ number of unknowns, which are solved using Newton's method. The results are obtained numerically with the help of MATHEMATICA software for $n = 7$.

2.7 Results and Discussion

In this section, the numerical values of solute concentration are displayed through Figs. 2.1-2.13 in presence/absence of advection and reaction terms with the variations of the temporal parameter α keeping spatial parameter β fixed and also with variations of β for fixed α.

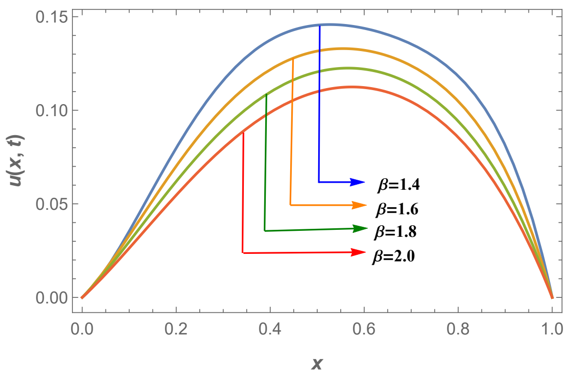

FIGURE 2.1: Variations of $u(x,t)$ vs. x for conservative system at $t = 0.6$ when v=0.2.

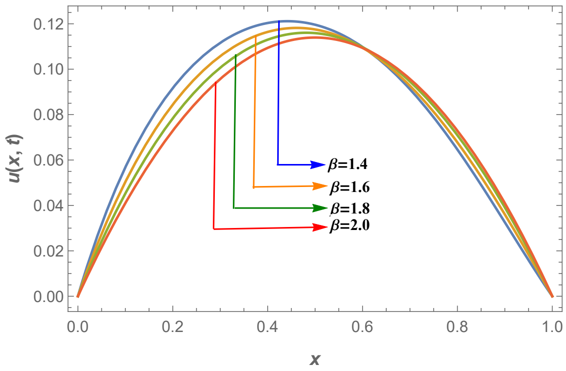

FIGURE 2.2: Variations of $u(x,t)$ vs. x for conservative system at $t = 0.6$ when v=0.

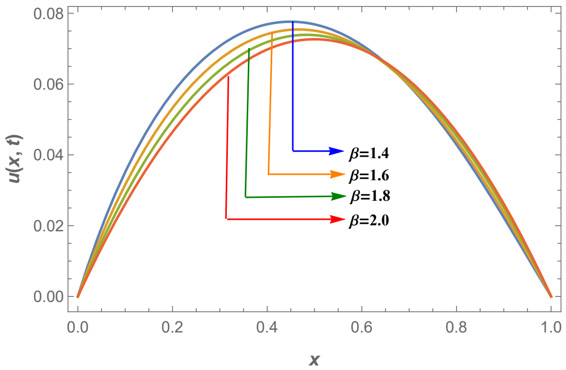

FIGURE 2.3: Variations of $u(x,t)$ vs. x for non-conservative system at $t = 0.6$ when v=0.

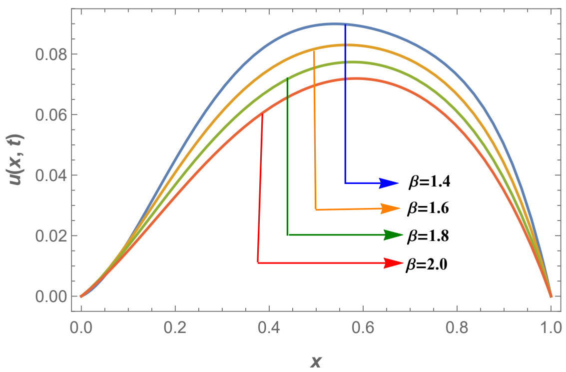

FIGURE 2.4: Variations of $u(x,t)$ vs. x for non-conservative system at $t = 0.6$ when v=0.2.

The variations of $u(x,t)$ vs. x at $t = 0.6$ are displayed through Figs. 2.1-2.2 and Figs. 2.3-2.4 for conservative system (k =0) and also for non-conservative system ($k = -1$) respectively keeping $\alpha = 1$ and β=1.4(0.2) 2.0, to show the effect of advection term. It is seen that in both the cases in presence of advection term, $u(x,t)$ decreases as β increases but afterwards the opposite nature occurs. It is also noticed that for non-conservative system the damping occurs due to presence of sink term ($k = -1$). Again for conservative and non-conservative systems the translations of the solute concentration are clearly visible with the fluid velocity ($v = 0.2$) without the changes of the slopes of the curves.

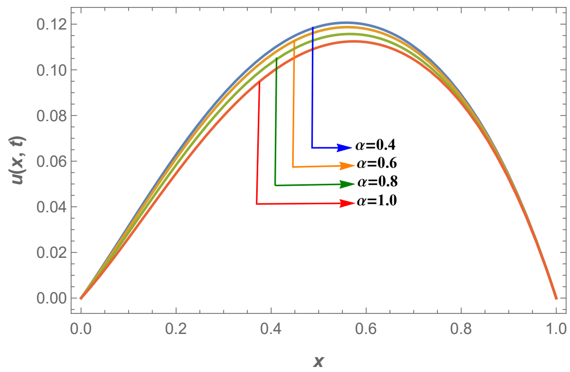

FIGURE 2.5: Variations of $u(x, t)$ vs. x for conservative system at $t = 0.6$ when v=0.2.

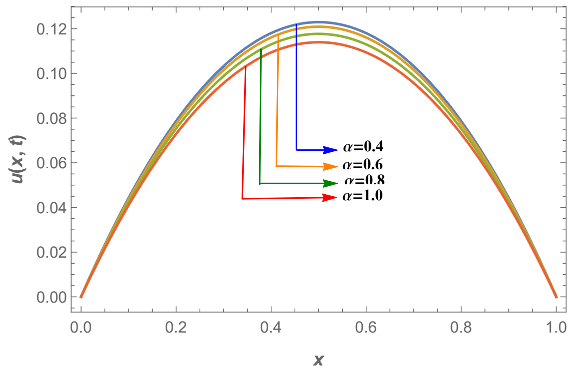

FIGURE 2.6: Variations of $u(x, t)$ vs. x for conservative system at $t = 0.6$ when v=0.

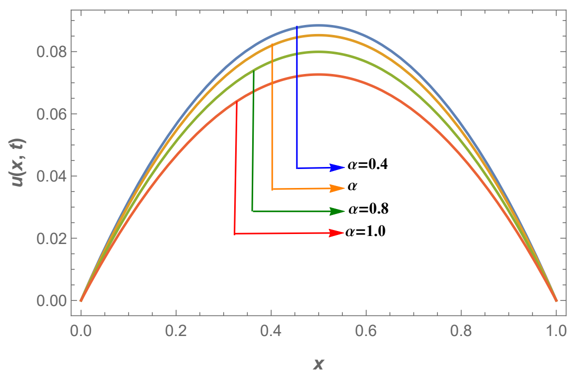

FIGURE 2.7: Variations of $u(x, t)$ vs. x for non-conservative system at $t = 0.6$ when v=0.

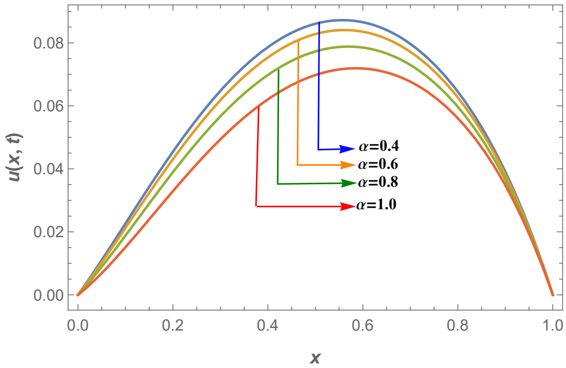

FIGURE 2.8: Variations of $u(x, t)$ vs. x for non-conservative system at $t = 0.6$ when v=0.2.

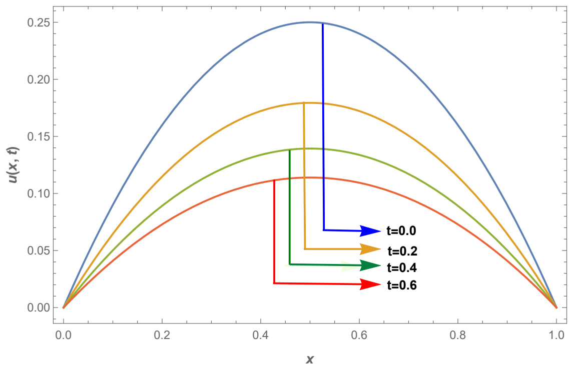

FIGURE 2.9: Variations of $u(x, t)$ vs. x for non-conservative system for different value of t when v=0.2.

Figs. 2.5-2.6 and Figs. 2.7-2.8 are drawn to show the variations of $u(x, t)$ with x at $t = 0.6$ for conservative and non-conservative systems taking $\beta = 1$ and α=0.4(0.2) 1.0. The nature of figures is similar to previous cases except changes in overshoots. Fig. 2.9 is drawn for $u(x, t)$ vs. x for various values of t to observe the effect on solute concentration due to increase in time. It is also observed that solute concentration is decreasing with the increase in time.

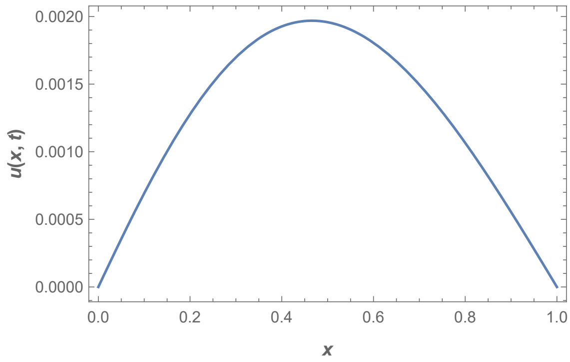

FIGURE 2.10: Variations of $u(x,t)$ vs. x for linear diffusive term at $\alpha = 1$, $\beta = 1.8$, $k = -1$, $v = 0.2$ at $t = 0.5$.

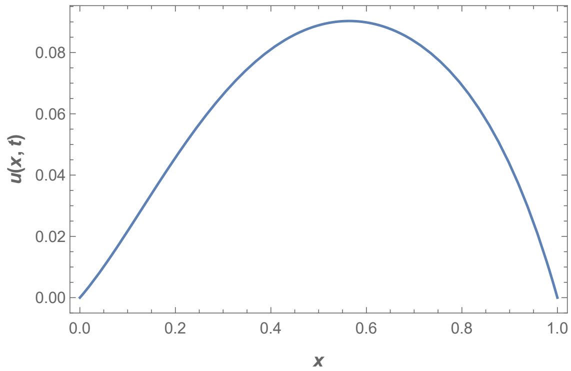

FIGURE 2.11: Variations of $u(x,t)$ vs. x for non-linear diffusive term at $\alpha = 1$, $\beta = 1.8$, $k = -1$, $v = 0.2$ at $t = 0.5$.

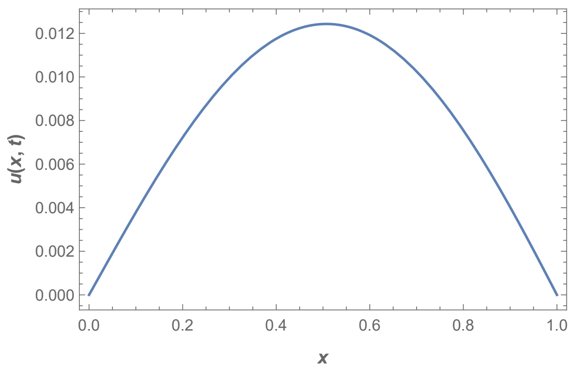

FIGURE 2.12: Numerical solution of $u(x,t)$ vs. x for linear diffusive term at $\alpha = 0.8$, $\beta = 2$, $k = -1$, $v = 0.2$ at $t = 0.5$.

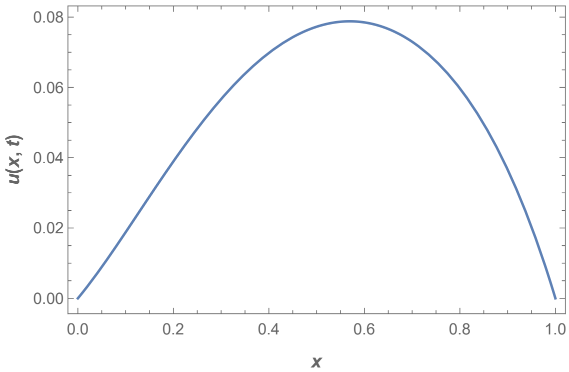

FIGURE 2.13: Numerical solution of $u(x,t)$ vs. x for non-linear diffusive term at $\alpha = 0.8$, $\beta = 2$, $k = -1$, $v = 0.2$ at $t = 0.5$.

Fig. 2.11 and Fig. 2.13 show the effects of non-linearity in diffusion term as compared to Fig. 2.10 and Fig. 2.12 at $t = 0.5$, $v = 0.2$, $k = -1$ for the cases $\alpha = 1$, $\beta = 1.8$, and $\beta = 2$, $\alpha = 0.8$ respectively. It is clear from the figures that there will be a lot of variations in the solute concentration due to presence of non-linearity in diffusion term.

2.8 Conclusion

In the present chapter, the collocation method is applied through defining the Fibonacci operational matrix in terms of Caputo fractional order derivatives during the solution of one-dimensional space-time fractional order ARDE with nonlinearity in the diffusive term. Numerical solutions are graphically presented for different particular cases keeping $\alpha = 1$ and varying spatial derivative β, and also varying temporal derivative α for $\beta = 2$. The salient feature of the study is the effect of reaction term on the solute concentration profile for both the cases in presence and absence of advection term. The decay rate of the solute concentration due to the sink term ($k = -1$) confirms the damping nature of the solution profile due to the presence of reaction term. The efficiency and effectiveness of the proposed method are validated by comparing the numerical solutions and the exact solutions of three existing problems through error analysis. The important feature of the article is the exhibition of convergence rate for two existing problems when the proposed method is applied. The effect of non-linearity in the diffusive term on the solution profile for different particular cases is the significant part of the study. The author believes that the present contribution will be beneficial for the researchers working in the field of the nonlinear diffusion equation in both integer and fractional orders.