Chapter 2

Approximation of Caputo–Prabhakar Derivative With Application in Solving Time Fractional Advection–Diffusion Equation

In this chapter, two numerical approximations for Caputo–Prabhakar derivative are discussed using linear and quadratic interpolation polynomials. Further, these schemes are combined with central difference approximation to solve time fractional advection–diffusion equation. Stability and solvability of the discussed schemes are studied. Finally, convergence results are verified numerically.

2.1 Introduction

This chapter deals with developing two numerical schemes namely, $NS1$ and $NS2$ using time stepping linear and quadratic interpolation polynomials to approximate the Caputo–Prabhakar derivative. Then utilizing these schemes we solve the time fractional advection–diffusion equation defined in the Caputo–Prabhakar sense using central difference approximation for the spatial derivatives. We study the stability analysis of these schemes using Fourier analysis. Further, we show that the proposed schemes are uniquely solvable. The convergence order of the whole discretized schemes $NS1$ and $NS2$ are $(\tau^{2-\alpha}, h^2)$ and $(\tau^{3-\alpha}, h^2)$, respectively. With the help of numerical experimentation the convergence results are validated numerically.

The rest of the chapter is organized as follows: in Section 2.2, we study the numerical approximation of the Caputo–Prabhakar derivative using two methods. The theoretical error bound of the two schemes is discussed in Section 2.3. Section 2.4 is devoted to the application of these two schemes for solving the time fractional advection–diffusion equation. Solvability and stability of both the schemes are also discussed in this section. In Section 2.5, we discuss numerical examples to validate our analytical results. Conclusions are included in the Section 2.6.

2.2 Approximation of Caputo–Prabhakar Derivative

Here, first we discretize the given interval $[0, T]$ into M, number of subintervals, each of length $\tau = \frac{T}{M}$ using $M + 1$ points $0 = t_0 < t_1 < t_2 ... < t_M = T$, where $t_j = j\tau, j = 0, 1, \dots M$. Then we approximate the Caputo–Prabhakar derivative in each subinterval using the following two schemes.

2.2.1 Numerical Scheme 1 $(NS1)$

In the numerical scheme 1 $(NS1)$, we approximate the function $f(t)$ by using time stepping linear interpolation polynomial. For two points $(t_{j-1}, f(t_{j-1}))$ and $(t_j, f(t_j))$ linear interpolation polynomial $P_{1,j}(t)$ and its derivative are defined as

$$\begin{aligned} P_{1,j}(t) =& \frac{t-t_{j-1}}{t_j-t_{j-1}} f(t_j) + \frac{t-t_j}{t_{j-1}-t_j} f(t_{j-1}), \\ P'_{1,j}(t) =& \frac{f(t_j)-f(t_{j-1})}{\tau}, \end{aligned} \tag{2.1}$$

and the error involved in approximating $f(t)$ by $P_{1,j}(t)$ is obtained as

$$f(t) - P_{1,j}(t) = \frac{f''(\xi_j)}{2}(t-t_j)(t-t_{j-1}), \tag{2.2}$$

where $\xi_j \in (t_{j-1}, t_j)$. For $0 < \alpha < 1$ and $0 < t < T$ Caputo–Prabhakar derivative is defined as

$${}_0^{\mathrm{CP}}D^{\gamma}_{\rho,\alpha,\omega} f(t) = \int_0^t (t-s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t-s)^{\rho}) f'(s) ds, \tag{2.3}$$

at point $t = t_k$ we have

$$\begin{aligned} {}_0^{\mathrm{CP}}D^{\gamma}_{\rho,\alpha,\omega} f(t_k) \approx & \int_0^{t_k} (t_k-s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-s)^{\rho}) P'_{1,j}(s) ds \\ =& \sum_{j=1}^{k} \int_{t_{j-1}}^{t_j} (t_k-s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-s)^{\rho}) P'_{1,j}(s) ds \\ =& \sum_{j=1}^{k} \int_{t_{j-1}}^{t_j} (t_k-s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-s)^{\rho}) \frac{f(t_j)-f(t_{j-1})}{\tau} ds \\ =& \tau^{-\alpha} \sum_{j=1}^{k} w_{1,k-j} f(t_j) + w_{2,k-j} f(t_{j-1}), \end{aligned}$$

where

$$w_{1,k-j} =(k-j+1)^{1-\alpha}E^{-\gamma}_{\rho,2-\alpha}(\omega(t_k-t_{j-1})^{\rho})-(k-j)^{1-\alpha}E^{-\gamma}_{\rho,2-\alpha}(\omega(t_k-t_j)^{\rho}), \quad (2.4)$$

$$w_{2,k-j} =(k-j)^{1-\alpha}E^{-\gamma}_{\rho,2-\alpha}(\omega(t_k-t_j)^{\rho})-(k-j+1)^{1-\alpha}E^{-\gamma}_{\rho,2-\alpha}(\omega(t_k-t_{j-1})^{\rho}). \quad (2.5)$$

Let $f(t_j)=f_j$ so we have,

$${}^{\mathrm{CP}}_{\;0}D^{\gamma}_{\rho,\alpha,\omega}f(t_k)=\tau^{-\alpha}\sum_{j=0}^{k}a_jf_{k-j}+R^k(f,\tau,\alpha). \quad (2.6)$$

For different values of k, the coefficients a_j are obtained as,

$$\begin{cases} a_0=w_{1,0}, \\ a_n=w_{1,n}+w_{2,n-1},\ \ 0<n<k \\ a_k=w_{2,k-1}. \end{cases}$$

In the following, we study some properties of the coefficients a_j in (2.6).

Lemma 2.2.1. For any $0<\alpha<1$, the following properties hold,

1. If we fix $k=1$, then $a_1=-a_0$,
2. $\sum_{j=0}^{k}a_j=0$.

Proof: (1) From the values of the coefficients a_j when $k=1$ and using (2.4) and (2.5),

$$a_0=w_{1,0}=E^{-\gamma}_{\rho,2-\alpha}(\omega\tau^{\rho}),$$

$$a_1=w_{2,0}=-E^{-\gamma}_{\rho,2-\alpha}(\omega\tau^{\rho}).$$

So, we have $a_1 = -a_0$.

(2) By using the values of the coefficients a_j and (2.4), (2.5) we have,

$$\begin{aligned}\sum_{j=0}^{k} a_j =& a_0 + a_1 + a_2 + \cdots + a_k \\ =& w_{1,0} + w_{1,1} + w_{2,0} + w_{1,2} + w_{2,2} + \cdots + w_{1,k-1} + w_{2,k-2} + w_{2,k-1} \\ =& (w_{1,0} + w_{2,0}) + (w_{1,1} + w_{2,1}) + \cdots + (w_{1,k-1} + w_{2,k-1}) \\ =& 0.\end{aligned}$$

The proof is completed.

2.2.2 Numerical Scheme 2 $(NS2)$

At grid point $t = t_k$, the Caputo–Prabhakar derivative of order $0 < \alpha < 1$ is defined as

$$\begin{aligned}{}_{0}^{\mathrm{CP}}D^{\gamma}_{\rho,\alpha,\omega} f(t_k) =& \int_0^{t_k} (t_k - s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k - s)^{\rho}) f'(s)ds \\ =& \sum_{j=1}^{k} \int_{t_{j-1}}^{t_j} (t_k - s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k - s)^{\rho}) f'(s)ds. \qquad (2.7)\end{aligned}$$

In the first subinterval $[t_0, t_1]$, we use linear interpolation polynomial to approximate the function $f(t)$ so,

$$\begin{aligned}\int_{t_0}^{t_1} (t_k - s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k - s)^{\rho}) f'(s)ds \approx& \int_{t_0}^{t_1} (t_k - s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k - s)^{\rho}) P'_{1,j}(s)ds \\ =& \frac{f_1 - f_0}{\tau} \int_{t_0}^{t_1} (t_k - s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k - s)^{\rho})ds \\ =& \tau^{-\alpha} b_k (f_1 - f_0), \qquad (2.8)\end{aligned}$$

where

$$b_k = k^{1-\alpha}E^{-\gamma}_{\rho,2-\alpha}(\omega(t_k)^\rho) - (k-1)^{1-\alpha}E^{-\gamma}_{\rho,2-\alpha}(\omega(t_k - t_1)^\rho). \tag{2.9}$$

And for other subintervals ($j \geq 2$), we use quadratic interpolation polynomial to approximate the function $f(t)$. For node points $(t_{j-2}, f(t_{j-2}))$, $(t_{j-1}, f(t_{j-1}))$ and $(t_j, f(t_j))$ quadratic interpolation polynomial $P_{2,j}(t)$ and its derivative are defined as

$$\begin{aligned} P_{2,j}(t) =& \frac{(t-t_{j-1})(t-t_j)}{2\tau^2}f_{j-2} - \frac{(t-t_{j-2})(t-t_j)}{\tau^2}f_{j-1} + \frac{(t-t_{j-2})(t-t_{j-1})}{2\tau^2}f_j, \\ P'_{2,j}(t) =& \frac{2t-t_{j-1}-t_j}{2\tau^2}f_{j-2} - \frac{2t-t_{j-2}-t_j}{\tau^2}f_{j-1} + \frac{2t-t_{j-2}-t_{j-1}}{2\tau^2}f_j, \end{aligned} \tag{2.10}$$

and the error involved in approximating $f(t)$ by $P_{2,j}(t)$ is obtained as

$$f(t) - P_{2,j}(t) = \frac{f'''(\xi_j)}{3!}(t-t_{j-2})(t-t_{j-1})(t-t_j), \tag{2.11}$$

where $\xi_j \in (t_{j-2}, t_j)$. Using $P_{2,j}(t)$ to approximate $f(t)$ in (2.7) for ($j \geq 2$), we have

$$\begin{aligned} &\sum_{j=2}^{k}\int_{t_{j-1}}^{t_j}(t_k - s)^{-\alpha}E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-s)^\rho)f'(s)ds \\ &\qquad\qquad \approx \sum_{j=2}^{k}\int_{t_{j-1}}^{t_j}(t_k - s)^{-\alpha}E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-s)^\rho)P'_{2,j}(s)ds \\ &\qquad\qquad = \tau^{-\alpha}\sum_{j=2}^{k}W_{1,k-j}f_j + W_{2,k-j}f_{j-1} + W_{3,k-j}f_{j-2}, \end{aligned} \tag{2.12}$$

where

$$W_{1,k-j} = \frac{1}{2}\Big[-(k-j)^{1-\alpha}\,{}_1E^k_{j+1} - (k-j+1)^{1-\alpha}\,{}_1E^k_j$$

$$
\begin{aligned}
&\quad - 2(k-j)^{2-\alpha}{}_2E_{j+1}^k + 2(k-j+1)^{2-\alpha}\;{}_2E_j^k\Big],\\
W_{2,k-j} =& 2\Big[(k-j)^{1-\alpha}{}_1E_{j+1}^k + (k-j)^{2-\alpha}{}_2E_{j+1}^k - (k-j+1)^{2-\alpha}{}_2E_j^k\Big],\\
W_{3,k-j} =& \frac{1}{2}\Big[-3(k-j)^{1-\alpha}{}_1E_{j+1}^k + (k-j+1)^{1-\alpha}\;{}_1E_j^k\\
&\quad - 2(k-j)^{2-\alpha}{}_2E_{j+1}^k + 2(k-j+1)^{2-\alpha}\;{}_2E_j^k\Big],\\
{}_1E_j^k =& E_{\rho,2-\alpha}^{-\gamma}(\omega(t_k - t_{j-1})^\rho),\\
{}_2E_j^k =& E_{\rho,3-\alpha}^{-\gamma}(\omega(t_k - t_{j-1})^\rho).
\end{aligned}
$$

Combining (2.8) and (2.12), we obtain an approximation scheme for Caputo–Prabhakar derivative,

$$
{}_0^{\mathrm{CP}}D_{\rho,\alpha,\omega}^{\gamma} f(t_k) = \tau^{-\alpha}\sum_{j=0}^{k} c_j f_{k-j} + E^k(f,\tau,\alpha), \tag{2.13}
$$

where values of the coefficients c_j for different values of k are given as follows:

For $k = 1$,

$$
\begin{cases}
c_0 = b_1,\\
c_1 = -b_1.
\end{cases}
$$

For $k = 2$,

$$
\begin{cases}
c_0 = W_{1,0},\\
c_1 = W_{2,0} + b_2,\\
c_2 = W_{3,0} - b_2.
\end{cases}
$$

For $k = 3$,

$$\begin{cases} c_0 = W_{1,0}, \\ c_1 = W_{2,0} + W_{1,1}, \\ c_2 = W_{3,0} + W_{2,1} + b_3, \\ c_3 = W_{3,1} - b_3. \end{cases}$$

For $k > 3$,

$$\begin{cases} c_0 = W_{1,0}, \\ c_1 = W_{2,0} + W_{1,1}, \\ c_n = W_{3,n-2} + W_{2,n-1} + W_{1,n}, \ 1 < n < k-1, \\ c_{k-1} = W_{3,k-3} + W_{2,k-2} + b_k, \\ c_k = W_{3,k-2} - b_k. \end{cases}$$

Now, we study some properties of the coefficients c_j in (2.13).

Lemma 2.2.2. For any $0 < \alpha < 1$, the following properties hold,

1. If we fix $k = 1$, then $c_1 = -c_0$.
2. $\sum_{j=0}^{k} c_j = 0$.

Proof: (1) If we fix $k = 1$, then the above scheme reduces to evaluate the Caputo–Prabhakar derivative of the function $f(t)$ at the node t_1. So $c_0 = b_1$ and $c_1 = -b_1$. Hence $c_0 = -c_1$.

(2) Here, we have

$$\sum_{j=0}^{k} c_j = c_0 + c_1 + c_2 + \cdots + c_{k-1} + c_k$$

$$= \sum_{j=0}^{k-2} W_{1,j} + W_{2,j} + W_{3,j} = 0.$$

2.3 Error Analysis of the Schemes

In the present section, we establish theoretical error bound for both the schemes.

2.3.1 Error Analysis of $NS1$

Theorem 2.3.1. Let $f(t) \in C^2[0,1]$ then for any $0 < \alpha < 1$, the truncation error $R^k(f,\tau,\alpha)$ is given as

$$|R^k(f,\tau,\alpha)| \leq C_1 \frac{\tau^{2-\alpha}}{2} \max_{0\leq t\leq t_k} \left|f''(t)\right|, \text{ where } C_1 \text{ is a constant.} \tag{2.14}$$

Proof.

$$\begin{aligned}
|R^k(f,\tau,\alpha)| &= \left|\sum_{j=1}^{k} \int_{t_{j-1}}^{t_j} (t_k - s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-s)^{\rho})(f - P_{1,j})'(s)ds\right| \\
&= \left|\sum_{j=1}^{k} \frac{f''(\xi_j)}{2} \int_{t_{j-1}}^{t_j} (s-t_j)(s-t_{j-1})(t_k-s)^{-\alpha-1} E^{-\gamma}_{\rho,-\alpha}(\omega(t_k-s)^{\rho})ds\right| \\
&\leq \left|\sum_{j=1}^{k} \frac{\tau^2}{8} f''(\xi_j)[(t_k-t_j)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-t_j)^{\rho}) \right. \\
&\quad \left. - (t_k - t_{j-1})^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-t_{j-1})^{\rho})]\right| \\
&\leq \left|\frac{\tau^{2-\alpha}}{8} \sum_{j=1}^{k} f''(\xi_j)[(k-j)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-t_j)^{\rho}) \right. \\
&\quad \left. - (k-j+1)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-t_{j-1})^{\rho})]\right| \\
&\leq C_1 \frac{\tau^{2-\alpha}}{8} \max_{0\leq t\leq t_k} \left|f''(t)\right|,
\end{aligned}$$

where C_1 is a constant. □

2.3.2 Error Analysis of $NS2$

Theorem 2.3.2. Let $f(t) \in C^3[0,1]$ then for any $0 < \alpha < 1$, error estimate of the numerical scheme 2 is given as

1. If $k = 1$,

$$|E^1(f,\tau,\alpha)| \leq C_1 \frac{\tau^{2-\alpha}}{8} \max_{0\leq t\leq t_1} \left|f''(t)\right|, \text{ where } C_1 \text{ is a constant.} \tag{2.15}$$

2. If $k \geq 2$

$$|E^k(f,\tau,\alpha)| \leq C_2 \frac{\tau^{3-\alpha}}{12} \max_{t_1\leq t\leq t_k} |f'''(t)|, \text{ where } C_2 \text{ is a constant.} \tag{2.16}$$

Proof. (1) If $k = 1$ then scheme (2.13) reduces to scheme $NS1$ and its convergence order is $O(\tau^{2-\alpha})$, which is proved in the previous theorem (2.3.1).
(2) For $k \geq 2$,

$$\begin{aligned}
E^k(f,\tau,\alpha) &= \left| \int_{t_1}^{t_k} (t_k - s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k - s)^{\rho})(f - P_{2,j})'(s)ds \right| \\
&= \left| \sum_{j=2}^{k} \int_{t_{j-1}}^{t_j} (t_k - s)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha} \left(\omega(t_k - s)^{\rho}\right) (f - P_{2,j})'(s)ds \right| \\
&= \left| \sum_{j=2}^{k} \int_{t_{j-1}}^{t_j} (t_k - s)^{-\alpha-1} E^{-\gamma}_{\rho,-\alpha}(\omega(t_k - s)^{\rho}) \frac{f'''(\xi_j)}{6} (s - t_{j-2})(s - t_{j-1})(s - t_j)ds \right| \\
&\leq \left| \frac{\tau^3}{12} \sum_{j=2}^{k} f'''(\xi_j) \int_{t_{j-1}}^{t_j} (t_k - s)^{-\alpha-1} E^{-\gamma}_{\rho,-\alpha}(\omega(t_k - s)^{\rho}) \right| \\
&\leq \left| \frac{\tau^{3-\alpha}}{12} \sum_{j=2}^{k} f'''(\xi_j) \left[(k - j)^{-\alpha} E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k - t_j)^{\rho}) \right.\right.
\end{aligned}$$

$$-(k-j+1)^{-\alpha}E^{-\gamma}_{\rho,1-\alpha}(\omega(t_k-t_{j-1})^{\rho})\Big]\Bigg|$$
$$\leq C_2\frac{\tau^{3-\alpha}}{12}\max_{t_1\leq t\leq t_k}|f'''(t)|,$$

where C_2 is a constant. □

2.4 Application

To describe solute transport in aquifers, the commonly used equation is the advection–diffusion equation. It is derived from the principle of conservation using Fick's law. The rate of change of any scalar quantity in any physical system is given by flow and diffusion, inside and out of that system, along with the source term, i.e.,

$$\frac{\partial u}{\partial t}+\nabla.j=R, \tag{2.17}$$

where j is the total flux, R is the source term, u describes mass, energy, or any other physical quantity. In this case, there are two sources of flux, namely, diffusive flux which arises due to diffusion and advective flux arises due to advection of particles. Diffusive flux is approximated by Fick's first law,

$$j_{diff}=-K\nabla u,$$

i.e., diffusive flux is proportional to the local concentration gradient in any part of the system, where, $K>0$ is the diffusion coefficient. Advective flux is approximated as,

$$j_{adv}=Vu,$$

where V is the average fluid velocity. Hence, the total flux is given as,

$$j = j_{adv} + j_{diff} = Vu - K\nabla u.$$

So, from (2.17),

$$\begin{aligned} \frac{\partial u}{\partial t} + \nabla.(Vu - K\nabla u) &= R, \\ \frac{\partial u}{\partial t} &= \nabla.(K\nabla u) - \nabla.(Vu) + R. \end{aligned} \tag{2.18}$$

Eq. (2.18) is the classical advection–diffusion equation. It reduces to a fractional advection–diffusion equation by taking one or more partial derivatives of fractional order. The fractional advection–diffusion equation was firstly proposed by Chaves [55] to investigate the mechanism of super diffusion and with the goal of having a model able to generate the Levy distribution.

In this section, we discuss the application of above-mentioned schemes in solving time fractional advection–diffusion equation,

$$ {}_{0}^{\mathrm{CP}}D^{\gamma}_{\rho,\alpha,\omega}u(x,t) = K\frac{\partial^2 u(x,t)}{\partial x^2} - V\frac{\partial u(x,t)}{\partial x} + F(x,t), \;\; 0 < x < L, 0 < t < T. \tag{2.19}$$

With the initial and boundary conditions,

$$u(x,0) = u_0(x), \tag{2.20}$$

$$\begin{cases} u(0,t) = g_1(t), \\ u(L,t) = g_2(t), \end{cases} \tag{2.21}$$

where $\alpha \in (0,1)$ denotes the order of the time fractional derivative. The given interval $[0,\ L]$ is divided into N number of subintervals each of length $h = \frac{L}{N}$ using $N+1$ points $0 = x_0 < x_1 < x_2 ... < x_N = L$, where $x_j = jh, j = 0, 1, \ldots, N$. We approximate the space derivatives of first and second order using central difference scheme as follows,

$$\frac{\partial u}{\partial x} = \frac{u_{i+1}^k - u_{i-1}^k}{2h} + \mathcal{O}(h^2), \tag{2.22}$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+1}^k - 2u_i^k + u_{i-1}^k}{h^2} + \mathcal{O}(h^2). \tag{2.23}$$

2.4.1 Numerical Scheme 1 $(NS1)$

Now using (2.6), (2.22), and (2.23) in (2.19) we get

$$\tau^{-\alpha} \sum_{j=0}^{k} a_{k-j} u_i^j = \frac{K}{h^2}(u_{i+1}^k - 2u_i^k + u_{i-1}^k) - \frac{V}{2h}(u_{i+1}^k - u_{i-1}^k) + F(x_i, t_k), \tag{2.24}$$

where $1 \leq k \leq M$ and $1 \leq i \leq N-1$. We can rewrite the initial and the boundary conditions as

$$u_i^0 = u_0(x_i), \qquad 0 \leq i \leq N$$

$$\begin{cases} u_0^k = g_1(t_k), & \\ u_N^k = g_2(t_k), & 1 \leq k \leq M \end{cases}$$

Let $h^2\tau^{-\alpha} = \mu$ so (2.24) can be rewritten as

$$\left(-K - \frac{Vh}{2}\right) u_{i-1}^k + (\mu a_0 + 2K)\, u_i^k + \left(-K + \frac{Vh}{2}\right) u_{i+1}^k$$
$$= -\mu \sum_{j=0}^{k-1} a_{k-j} u_i^j + h^2 F(x_i, t_k). \tag{2.25}$$

Eq. (2.25) can be written in the form of a matrix as

$$AU^k = -\mu \sum_{j=0}^{k-1} a_{k-j} U^j + h^2 F^k + B^k, \tag{2.26}$$

where the matrix and the vectors in (2.26) are given as

$$A = \text{tri}\left[-K - \tfrac{Vh}{2}, \mu a_0 + 2K, -K + \tfrac{Vh}{2}\right],$$

$$U^k = \left[u_1^k, u_2^k, \ldots, u_{N-1}^k\right]^T,$$

$$F^k = \left[F_1^k, F_2^k, \ldots, F_{N-1}^k\right]^T,$$

$$B^k = \left[\left(K + \tfrac{Vh}{2}\right) u_0^k, 0, \ldots, 0, \left(K - \tfrac{Vh}{2}\right) u_N^k\right]^T, \qquad 1 \le k \le M.$$

Now, by using Fourier method [97] we will study the stability analysis of the above scheme $NS1$. Let $\hat{u}_i^k$ be the approximate solution of (2.25) and let us define

$$\phi_i^k = u_i^k - \hat{u}_i^k, \ 1 \le i \le N-1, \ 1 \le k \le M,$$

with the corresponding vector

$$\phi^k = \left[\phi_1^k, \phi_2^k, \ldots, \phi_{N-1}^k\right]^T.$$

So, we obtain the following error equations

$$\left(-K - \frac{Vh}{2}\right) \phi_{i-1}^k + (\mu a_0 + 2K)\, \phi_i^k + \left(-K + \frac{Vh}{2}\right) \phi_{i+1}^k = -\mu \sum_{j=0}^{k-1} a_{k-j} \phi_i^j. \tag{2.27}$$

For the Fourier method, the grid functions are defined as

$$\phi^k(x) = \begin{cases} \phi_i^k, & x_i - \frac{h}{2} < x \leq x_i + \frac{h}{2}, \quad i = 1, 2, \ldots, N-1, \\ 0, & 0 \leq x \leq \frac{h}{2} \ \text{ or } \ L - \frac{h}{2} < x \leq L, \end{cases} \tag{2.28}$$

where $k = 1, 2, \ldots, M$. $\phi^k(x)$ can be expanded in a Fourier series as

$$\phi^k(x) = \sum_{l=-\infty}^{\infty} d_k(l) e^{I2\pi lx/L}, \quad k = 1, 2, ...M, \tag{2.29}$$

where

$$d_k(l) = \frac{1}{L} \int_0^L \phi^k(x) e^{-I2\pi lx/L} dx, \qquad I^2 = -1. \tag{2.30}$$

Now by using the definition of the discrete 2-norm

$$|\phi^k|_2 = \left(\sum_{i=1}^{N-1} h |\phi_i^k|^2 \right)^{1/2} = \left[\int_0^L |\phi^k(x)|^2 dx \right]^{1/2}. \tag{2.31}$$

By using Parseval equality

$$\int_0^L |\phi^k(x)|^2 dx = \sum_{l=-\infty}^{\infty} |d_k(l)|^2, \tag{2.32}$$

we will get

$$|\phi^k|_2^2 = \sum_{l=-\infty}^{\infty} |d_k(l)|^2. \tag{2.33}$$

Based on which, we can assume that solution of the (2.27) is of the form $\phi_i^k = d_k e^{I\sigma i h}$, where $\sigma = 2\pi l/L$. Putting the above expression in (2.27), we get

$$d_k = \frac{-\mu}{\left(4K\sin^2(\frac{\sigma h}{2}) + IVh\sin(\sigma h) + \mu a_0\right)} \sum_{j=0}^{k-1} a_{k-j} d_j. \tag{2.34}$$

Proposition 2.4.1. If d_k is the solution of (2.34), $(k = 1, 2, \ldots, M)$ then we have

$$|d_k| \leq |d_0|, \;\; k = 1, 2, \ldots, M.$$

Proof. We will prove the Proposition 2.4.1 by using the idea of mathematical induction. We notice that

$$\begin{aligned}\left|4K\sin^2\left(\frac{\sigma h}{2}\right) + IVh\sin(\sigma h) + \mu a_0\right| &= \sqrt{\left(4K\sin^2\left(\frac{\sigma h}{2}\right) + \mu a_0\right)^2 + V^2h^2\sin^2(\sigma h)}, \\ &\geq \sqrt{\mu^2 a_0^2} = \mu a_0.\end{aligned}$$

Now, for $k = 1$,

$$\begin{aligned}|d_1| &= \left|\frac{-\mu a_1 d_0}{\left(4K\sin^2\left(\frac{\sigma h}{2}\right) + IVh\sin(\sigma h) + \mu a_0\right)}\right| \\ &\leq \left|\frac{-\mu a_1 d_0}{\mu a_0}\right|,\end{aligned}$$

By using property 1 of Lemma 2.2.1, we know that $a_1 = -a_0$ so we have

$$|d_1| \leq |d_0|.$$

Now, suppose $|d_j| \leq |d_0|$ for $2 \leq j \leq k-1$ so we have from (2.27),

$$\begin{aligned}
|d_k| &= \left| \frac{-\mu}{\left(4K\sin^2\left(\frac{\sigma h}{2}\right) + IVh\sin(\sigma h) + \mu a_0\right)} \sum_{j=0}^{k-1} a_{k-j} d_j \right|, \\
&\leq \left| \frac{-\mu d_0}{\left(4K\sin^2\left(\frac{\sigma h}{2}\right) + IVh\sin(\sigma h) + \mu a_0\right)} \sum_{j=1}^{k} a_j \right|,
\end{aligned}$$

By using property 2 of Lemma 2.2.1 we have

$$\begin{aligned}
|d_k| =& \frac{\mu|d_0|}{|4K\sin^2\left(\frac{\sigma h}{2}\right) + IVh\sin(\sigma h) + \mu a_0|} \sum_{j=1}^{k} |a_j|, \\
\leq& \frac{\mu a_0|d_0|}{|4K\sin^2\left(\frac{\sigma h}{2}\right) + IVh\sin(\sigma h) + \mu a_0|}, \\
\leq& \frac{\mu a_0|d_0|}{\mu a_0},
\end{aligned}$$

so we have $|d_k| \leq |d_0|$. □

Now using the Proposition 2.4.1 in (2.33) we have

$$|\phi^k|_2^2 = \sum_{l=-\infty}^{\infty} |d_k(l)|^2 \leq \sum_{l=-\infty}^{\infty} |d_0(l)|^2 = |\phi^0|_2^2,$$

hence we have

$$|\phi^k|_2 \leq |\phi^0|_2, \; k = 1, 2, \ldots, M,$$

which proves the stability of the discussed scheme.

2.4.2 Numerical Scheme 2 $(NS2)$

Now, for numerical scheme 2, using (2.22), (2.23) and (2.13) in (2.19) we get,

$$\tau^{-\alpha}\left[c_0u_i^k + c_1u_i^{k-1} + \sum_{j=2}^{k-2} c_{k-j}u_i^j + c_{k-1}u_i^1 + c_ku_i^0\right]$$
$$= \frac{K}{h^2}(u_{i+1}^k - 2u_i^k + u_{i-1}^k) - \frac{V}{2h}(u_{i+1}^k - u_{i-1}^k) + F(x_i, t_k), \tag{2.35}$$

where $1 \leq k \leq M$ and $1 \leq i \leq N-1$. We can rewrite the initial and the boundary conditions as,

$$u_i^0 = u_0(x_i),\ 0 \leq i \leq N,$$

$$\begin{cases} u_0^k = g_1(t_k), \\ u_N^k = g_2(t_k),\ 1 \leq k \leq M. \end{cases}$$

Let $h^2\tau^{-\alpha} = \mu$ so (2.35) can be written as

$$\begin{cases} \left(-K - \frac{Vh}{2}\right)u_{i-1}^1 + (\mu c_0 + 2K)\,u_i^1 + \left(-K + \frac{Vh}{2}\right)u_{i+1}^1 \\ = -\mu c_1 u_i^0 + h^2F(x_i, t_1),\ k = 1, \\ \left(-K - \frac{Vh}{2}\right)u_{i-1}^k + (\mu c_0 + 2K)\,u_i^k + \left(-K + \frac{Vh}{2}\right)u_{i+1}^k \\ = -\mu c_1 u_i^{k-1} - \mu\sum_{j=2}^{k-2} c_{k-j}u_i^j - \mu(c_{k-1}u_i^1 + c_ku_i^0) + h^2F(x_i, t_k),\ k \geq 2, \end{cases} \tag{2.36}$$

so (2.36) can be written in the form of a matrix as,

$$\begin{cases} CU^1 = & -\mu c_1 U^0 + h^2 F^1 + B^1,\ k = 1, \\ CU^k = & -\mu c_1 U^{k-1} - \mu \sum_{j=2}^{k-2} c_{k-j} U^j - \mu \left(c_{k-1} U^1 + c_k U^0 \right) \\ & +h^2 F^k + B^k,\ k \geq 2. \end{cases} \tag{2.37}$$

The matrix and the vectors in (2.37) are defined as,

$$C = \text{tri}\left[-K - \tfrac{Vh}{2}, \mu c_0 + 2K, -K + \tfrac{Vh}{2} \right],$$

$$U^k = \left[u_1^k, u_2^k, \ldots, u_{N-1}^k \right]^T,$$

$$F^k = \left[F_1^k, F_2^k, \ldots, F_{N-1}^k \right]^T,$$

$$B^k = \left[\left(K + \tfrac{Vh}{2} \right) u_0^k, 0, \ldots, 0, \left(K - \tfrac{Vh}{2} \right) u_N^k \right]^T, \qquad 1 \leq k \leq M.$$

Now by using Fourier method, we will study the stability analysis of the scheme $NS2$. Let $\hat{u}_i^k$ be the approximate solution of (2.36) and let us define,

$$\phi_i^k = u_i^k - \hat{u}_i^k,\ 1 \leq i \leq N-1,\ 1 \leq k \leq M,$$

with the corresponding vector

$$\phi^k = \left[\phi_1^k, \phi_2^k, \ldots, \phi_{N-1}^k \right]^T.$$

So, we obtain the following error equations

$$\begin{cases} \left(-K-\frac{Vh}{2}\right)\phi_{i-1}^{1}+\left(\mu c_0+2K\right)\phi_i^1+\left(-K+\frac{Vh}{2}\right)\phi_{i+1}^{1} \\ =-\mu c_1\phi_i^0,\ k=1, \\ \left(-K-\frac{Vh}{2}\right)u_{i-1}^{k}+\left(\mu c_0+2K\right)u_i^k+\left(-K+\frac{Vh}{2}\right)u_{i+1}^{k} \\ =-\mu c_1\phi_i^{k-1}-\mu\sum_{j=2}^{k-2}c_{k-j}\phi_i^j-\mu(c_{k-1}\phi_i^1+c_k\phi_i^0),\ k\geq 2. \end{cases} \tag{2.38}$$

To study the stability analysis of the scheme by using Fourier method, we consider equations (2.28)-(2.33) and based on which we can assume that the solution of the (2.38) is of the form $\phi_i^k=d_ke^{I\sigma ih}$, where $\sigma=2\pi l/L$. Putting the above expression in (2.38) we get

$$\begin{cases} d_1=\frac{-\mu c_1 d_0}{\left(4K\sin^2\left(\frac{\sigma h}{2}\right)+IVh\sin(\sigma h)+\mu c_0\right)},\ k=1, \\ d_k=\frac{-\mu}{\left(4K\sin^2\left(\frac{\sigma h}{2}\right)+IVh\sin(\sigma h)+\mu c_0\right)}\left(c_1d_{k-1}+\sum_{j=2}^{k-2}c_{k-j}d_j-c_{k-1}d_1-c_kd_0\right),\ k\geq 2, \end{cases} \tag{2.39}$$

Proposition 2.4.2. If d_k is the solution of (2.39) for $(k=1,2\ldots M)$ then we have $|d_k|\leq|d_0|,\ k=1,2,\ldots M$.

Proof. We will prove the Proposition 2.4.2 by using the idea of mathematical induction. Notice that,

$$\begin{aligned}\left|4K\sin^2\left(\frac{\sigma h}{2}\right)+IVh\sin(\sigma h)+\mu c_0\right| &= \sqrt{\left(4K\sin^2\left(\frac{\sigma h}{2}\right)+\mu c_0\right)^2+V^2h^2\sin^2(\sigma h)}, \\ &\geq\sqrt{\mu^2c_0^2}=\mu c_0.\end{aligned}$$

Now for $k = 1$

$$|d_1| = \left|\frac{-\mu c_1 d_0}{4D\sin^2\left(\frac{\sigma h}{2}\right) + IVh\sin(\sigma h) + \mu c_0}\right|, \\ \leq \left|\frac{-\mu c_1 d_0}{\mu c_0}\right|.$$

By using the property 1 of the Lemma 2.2.2 we have

$$|d_1| \leq |d_0|.$$

Now let $|d_j| \leq |d_0|$ for $(1 < j < k)$ then we have,

$$|d_k| = \left|\frac{-\mu}{4K\sin^2\left(\frac{\sigma h}{2}\right) + IVh\sin(\sigma h) + \mu c_0}\left(c_1 d_{k-1} + \sum_{j=2}^{k-2} c_{k-j}d_j - c_{k-1}d_1 - c_k d_0\right)\right|, \\ \leq \frac{\mu|d_0|}{|4K\sin^2\left(\frac{\sigma h}{2}\right) + IVh\sin(\sigma h) + \mu c_0|}\sum_{j=1}^{k}|c_j|,$$

by using the Property 2 of the Lemma 2.2.2 we have,

$$|d_k| \leq \frac{\mu|d_0|c_0}{\mu c_0},$$

so we have $|d_k| \leq |d_0|$. □

Now applying the proposition 2.4.2 in (2.33) we get

$$|\phi^k|_2^2 = \sum_{l=-\infty}^{\infty}|d_k(l)|^2 \leq \sum_{l=-\infty}^{\infty}|d_0(l)|^2 = |\phi^0|_2^2,$$

so $|\phi^k|_2 \leq |\phi^0|_2,\ k = 1, 2, \ldots, M$, which proves the stability of the discussed scheme.

Lemma 2.4.1. Coefficient matrices A and C obtained in the schemes $NS1$, $NS2$ respectively are invertible.

Proof: [98] The eigenvalues of a tridiagonal matrix of the form, tri$[c\ a\ b]$ are as follows,

$$\lambda_i = a + 2b\left(\frac{c}{b}\right)^{1/2}\cos\left(\frac{i\pi}{N}\right),\ \text{where,}\ i = 1\ldots N-1.$$

So, we have for matrix $A = \left[-K - \frac{Vh}{2}, \mu a_0 + 2K, -K + \frac{Vh}{2}\right]$,

$$\begin{aligned}\lambda_i &= \mu a_0 + 2K + 2\sqrt{\left(-K - \frac{Vh}{2}\right)\left(-K + \frac{Vh}{2}\right)}\cos\left(\frac{i\pi}{N}\right),\\ &= \mu a_0 + 2K + 2\sqrt{K^2 - \frac{V^2h^2}{4}}\cos\left(\frac{i\pi}{N}\right), \quad i = 1\ldots N-1.\end{aligned}$$

For the case $K^2 - \frac{V^2h^2}{4} \leq 0$ the eigenvalues of A can be written as $\lambda_i = a + Ib$, where at least $a \neq 0$ since $K > 0$, $\mu = h^2\tau^{-\alpha} > 0$ and $a_0 = W_{1,0} \neq 0$ if $\omega \neq 0$.
Again for the case, $K^2 - \frac{V^2h^2}{4} > 0$, we have,

$$\lambda_i \geq \mu a_0 + 4D > 0,\ \text{if}\ \omega > 0.$$

So in each cases, eigenvalues of A are nonzero and hence the coefficient matrix A is invertible. Similarly, we can show that the matrix $C = \left[-K - \frac{Vh}{2},\ \mu c_0 + 2K, -K + \frac{Vh}{2}\right]$ is also invertible.

Theorem 2.4.1. The numerical schemes $NS1$ and $NS2$ have unique solutions.

Proof. In both the numerical schemes, to obtain the solution at each time level t_k, we have to solve the corresponding linear system of algebraic equations. Since the coefficient matrix in both the schemes is invertible, the solutions of the finite difference schemes $NS1$ and $NS2$ exist and are unique. □

2.5 Numerical Experiments

In the present section, we discuss some numerical examples to validate our analytical findings. To check the numerical accuracy, the maximum absolute error (MAE) between the exact and approximate solutions is defined as

$$\text{MAE} = \max_{0\leq i\leq N, 0\leq k\leq M} \left| U(x_i, t_k) - u_i^k \right|, \tag{2.40}$$

where $U(x_i, t_k)$ is the analytical solution and u_i^k is numerical solution. Convergence order (CO) is calculated using the following formula,

$$\text{CO}(j) = \frac{\log\big(\text{MAE}(j)/\text{MAE}(j+1)\big)}{\log\big(\tau(j)/\tau(j+1)\big)}, \tag{2.41}$$

$$\text{CO}(j) = \frac{\log\big(\text{MAE}(j)/\text{MAE}(j+1)\big)}{\log\big(h(j)/h(j+1)\big)}, \tag{2.42}$$

in the temporal and the spatial directions, respectively.

If the analytical solution is not known, we use the following method to calculate the convergence order. Let u_τ and $u_{\tau/2}$ be the approximated solutions calculated at the step sizes τ, $\tau/2$ respectively, and u_E is the exact solution. To calculate convergence order R, we perform the following error calculations:

$$e_\tau = u_E - u_\tau = C(\tau)^R, \tag{2.43}$$

$$e_{\frac{\tau}{2}} = u_E - u_{\frac{\tau}{2}} = C\left(\frac{\tau}{2}\right)^R, \tag{2.44}$$

where C is a constant. Subtracting Eq. (2.43) and (2.44), we get,

$$\Delta e_\tau = e_\tau - e_{\frac{\tau}{2}} = u_{\frac{\tau}{2}} - u_\tau = C(\tau)^R\left(1 - \frac{1}{2^R}\right). \tag{2.45}$$

Similarly, we have

$$\Delta e_{\frac{\tau}{2}} = e_{\frac{\tau}{2}} - e_{\frac{\tau}{4}} = u_{\frac{\tau}{4}} - u_{\frac{\tau}{2}} = C\left(\frac{\tau}{2}\right)^{R}\left(1 - \frac{1}{2^{R}}\right). \tag{2.46}$$

From Eq. (2.45) and (2.46), we get

$$R = \log_2\left(\frac{\Delta e_{\tau}}{\Delta e_{\frac{\tau}{2}}}\right). \tag{2.47}$$

Example 2.5.1. Calculate the αth order Caputo–Prabhakar fractional derivative of the function $f(t) = t^4$, $t \in [0,1]$. Since $f \in \mathrm{L}^1[0,1]$ so we can calculate its Caputo–Prabhakar derivative. The Caputo–Prabhakar derivative of the above function is $24t^{4-\alpha}E^{-\gamma}_{\rho,5-\alpha}(\omega t^{\rho})$. We have calculated MAE and CO, for the above function using both the numerical schemes $NS1$ and $NS2$ for different values of α. Obtained results are presented through the Table 2.1 and Table 2.2. From Table 2.1 and Table 2.2, it is clear that the convergence order for $NS1$ is $(2-\alpha)$ and for $NS2$ it is $(3-\alpha)$, which validate our analytical results.

Table 2.1: MAE and CO of the scheme $NS1$ for Ex. 2.5.1 for different values of α and τ.

α	τ	$NS1$		
		MAE	CO	CPU time (Sec.)
0.2	1/100	2.92662E-04		0.661859
	1/200	8.86059E-05	1.723757659	3.102035
	1/400	2.65645E-05	1.737904658	12.600272
	1/800	7.90371E-06	1.748897714	51.626952
	1/1200	3.87802E-06	1.756025052	143.387995
	1/1500	2.61873E-06	1.759569872	301.067124
0.4	1/100	1.47581E-03		0.994258
	1/200	4.96921E-04	1.570416172	4.533831
	1/400	1.66173E-04	1.580328503	18.26506
	1/800	5.53033E-05	1.587249740	60.01236
	1/1200	2.90092E-05	1.591304879	142.68601
	1/1500	2.03302E-05	1.593166031	271.450103
0.6	1/100	5.71319E-03		0.912751
	1/200	2.17575E-03	1.392781457	3.332103
	1/400	8.25247E-04	1.398617328	13.174441
	1/800	3.12264E-04	1.402056679	56.081137
	1/1200	1.76740E-04	1.403741915	143.757483
	1/1500	1.29192E-04	1.404400923	280.451740

TABLE 2.2: MAE and CO of the scheme $NS2$ for Ex. 2.5.1 for different values of α and τ.

α	τ	$NS2$		
		MAE	CO	CPU time (Sec.)
0.2	1/100	3.32797E-06		1.509844
	1/200	4.99005E-07	2.737515094	6.378224
	1/400	7.41959E-08	2.749644469	26.240775
	1/800	1.09623E-08	2.758787851	106.649324
	1/1200	3.57367E-09	2.764400993	286.094373
	1/1500	1.92738E-09	2.766964638	565.547859
0.4	1/100	1.71402E-05		1.160869
	1/200	2.86770E-06	2.579412678	5.817802
	1/400	4.77128E-07	2.587447205	23.426649
	1/800	7.90924E-08	2.592764782	93.941174
	1/1200	2.76090E-08	2.595720855	252.357765
	1/1500	1.54659E-08	2.596983478	517.774211
0.6	1/100	6.88032E-05		1.219815
	1/200	1.30627E-05	2.397022839	6.074419
	1/400	2.47207E-06	2.401662957	25.479885
	1/800	4.67021E-07	2.404159009	102.508335
	1/1200	1.76113E-07	2.405259100	276.217813
	1/1500	1.02958E-07	2.405628640	582.855793

Example 2.5.2. For the function $f(t) = t^{5-\alpha}$, calculate the αth order fractional derivative for $t \in [0, 1]$. Prabhakar derivative of the function $t^{5-\alpha}$ is not known. So we have calculated αth order derivative of this function using the formula (2.47). Obtained error and convergence order for both the schemes $NS1$ and $NS2$ are given through the Table 2.3 and Table 2.4. Results obtained validate the analytical findings.

TABLE 2.3: CO of the scheme $NS1$ for Ex. 2.5.2 using different values of α.

α	t	CO		
		$\log_2\left(\frac{\Delta e_\tau}{\Delta e_{\tau/2}}\right)$	$\log_2\left(\frac{\Delta e_{\tau/2}}{\Delta e_{\tau/4}}\right)$	$\log_2\left(\frac{\Delta e_{\tau/4}}{\Delta e_{\tau/8}}\right)$
	0.2	1.145591821	1.409888039	1.543884948
	0.4	1.410104554	1.544091165	1.618865872
0.2	0.6	1.498509569	1.592663892	1.648160849
	0.8	1.544295193	1.619067451	1.664793781
	1.0	1.572939753	1.636159557	1.675885071
	0.2	1.014117779	1.277808155	1.408941689
	0.4	1.278476741	1.409556659	1.480593993
0.4	0.6	1.365586946	1.456267893	1.507696732
	0.8	1.410220751	1.481222984	1.522700559
	1.0	1.437875606	1.497133359	1.532498341
	0.2	0.876607812	1.133412230	1.257204319
	0.4	1.134280638	1.258014377	1.322182468
0.6	0.6	1.217168613	1.300767604	1.345619003
	0.8	1.258895384	1.323019593	1.358132967
	1.0	1.284355446	1.336893599	1.366066486

Table 2.4: CO of scheme $NS2$ for Ex. 2.5.2 for different values of α.

α	t	CO		
		$\log_2\left(\frac{\Delta e_\tau}{\Delta e_{\tau/2}}\right)$	$\log_2\left(\frac{\Delta e_{\tau/2}}{\Delta e_{\tau/4}}\right)$	$\log_2\left(\frac{\Delta e_{\tau/4}}{\Delta e_{\tau/8}}\right)$
	0.2	1.926777009	2.347817803	2.533955845
	0.4	2.348023712	2.534157243	2.627141891
0.2	0.6	2.473289976	2.595617146	2.660680395
	0.8	2.534357753	2.627343254	2.678826729
	1.0	2.571055923	2.647142888	2.690523951
	0.2	1.837203137	2.237423165	2.407269559
	0.4	2.238054589	2.407869177	2.490298157
0.4	0.6	2.353135259	2.462914597	2.519391365
	0.8	2.408516611	2.490921934	2.534800326
	1.0	2.441495689	2.508185004	2.544552217
	0.2	1.741661592	2.111207336	2.260299289
	0.4	2.112038940	2.261093563	2.330571024
0.6	0.6	2.213849429	2.308031083	2.354236118
	0.8	2.261957029	2.331397972	2.366392095
	1.0	2.290215504	2.345537160	2.373889119

Example 2.5.3. Consider the advection–diffusion equation,

$${}_{0}^{\mathrm{CP}}D^{\gamma}_{\rho,\alpha,\omega}u(x,t) = \frac{\partial^2 u(x,t)}{\partial x^2} - \frac{\partial u}{\partial x} + F(x,t),\ 0 < x < 1, 0 < t < 1,$$

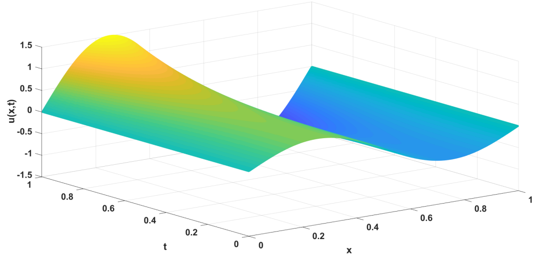
with the non homogeneous initial condition $u(x,0) = \alpha\sin(2\pi x)$ and boundary conditions $u(0,t) = u(1,t) = 0$ and $F(x,t) = 6\sin(2\pi x)t^{3-\alpha}E^{-\gamma}_{\rho,4-\alpha}(\omega t^{\rho}) + 4\pi^2(t^3 + \alpha)\sin(2\pi x) + 2\pi(t^3+\alpha)\cos(2\pi x)$. The analytical solution of this problem is $u(x,t) = (t^3+\alpha)\sin(2\pi x)$. The values of MAE and CO for different values of α are presented through the Table 2.5 and Table 2.6, for the schemes $NS1$ and $NS2$, respectively. From Table 2.5 and Table 2.6, it can be observed that the convergence order for both the schemes is of order 2. The analytical solution can be compared with the numerical solution from Fig. 2.1, when α is 0.5. From solution plot 2.1, it is clear that the numerical and analytical solutions are in good agreement.

TABLE 2.5: MAE and CO of the scheme $NS1$ for Ex. 2.5.3 for different values of α and τ.

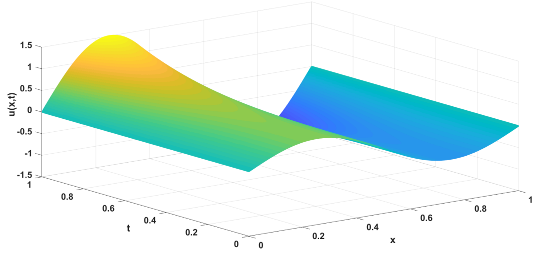
α	h	$NS1$		
		MAE	CO	CPU time (Sec.)
0.2	1/100	4.50798E-04		3.645007
	1/200	1.12852E-04	1.998054190	17.806968
	1/400	2.82569E-05	1.997752368	81.006717
	1/800	7.07674E-06	1.997444915	352.212202
	1/1200	3.14887E-06	1.997142445	935.372234
0.4	1/100	5.34626E-04		3.088415
	1/200	1.34917E-04	1.986463705	14.504813
	1/400	3.41481E-05	1.982189005	79.291354
	1/800	8.67489E-06	1.976889615	321.377755
	1/1200	3.89997E-06	1.971720889	679.395023
0.6	1/100	6.49852E-04		3.019408
	1/200	1.69976E-04	1.934781479	15.926448
	1/400	4.53443E-05	1.906335445	85.925916
	1/800	1.24166E-05	1.868657049	338.533437
	1/1200	5.90745E-06	1.832010987	908.404759

TABLE 2.6: MAE and CO of the scheme $NS2$ for Ex. 2.5.3 using different values of α and τ.

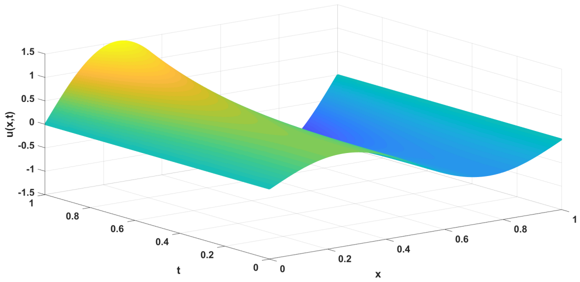
α	h	$NS2$		
		MAE	CO	CPU time (Sec.)
0.2	1/100	4.47746E-04		8.572939
	1/200	1.11943E-04	1.999920562	65.312789
	1/400	2.79843E-05	2.000069191	406.491634
	1/800	6.99615E-06	1.999986149	4951.744822
	1/1200	3.10939E-06	2.000008714	10835.688543
0.4	1/100	5.19559E-04		6.981035
	1/200	1.29892E-04	1.999970313	56.558576
	1/400	3.24708E-05	2.000104475	602.996818
	1/800	8.11764E-06	2.000009297	6567.663821
	1/1200	3.60782E-06	2.000015624	11962.908772
0.6	1/100	5.92169E-04		11.952338
	1/200	1.48033E-04	2.000090500	73.192005
	1/400	3.70031E-05	2.000200684	802.383659
	1/800	9.25019E-06	2.000090764	9095.776082
	1/1200	4.11104E-06	2.000096397	14562.540287



(a) Numerical solution using $NS1$



(b) Numerical solution using $NS2$



(c) Analytical solution

FIGURE 2.1: Comparison between numerical and analytical solution for Ex. 2.5.3 for $\alpha = 0.5$.

Example 2.5.4. Consider the advection–diffusion equation,

$$ {}_{0}^{\mathrm{CP}}D^{\gamma}_{\rho,\alpha,\omega}u(x,t) = \frac{\partial^2 u(x,t)}{\partial x^2} - \frac{\partial u}{\partial x} + F(x,t),\ 0 < x < 1, 0 < t < 1, $$

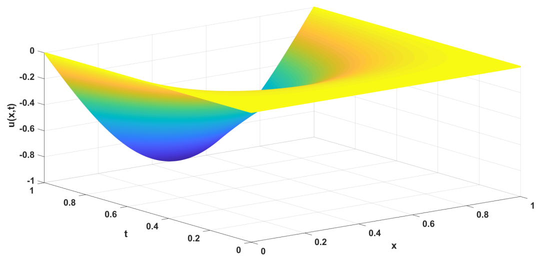
having the homogeneous initial condition $u(x,0) = 0$, and boundary conditions $u(0,t) = u(1,t) = 0$ and $F(x,t) = 2\cos\big(\pi x + \frac{\pi}{2}\big)t^{2-\alpha}E^{-\gamma}_{\rho,3-\alpha}(\omega t^{\rho}) + \pi^2 t^2 \cos\big(\pi x + \frac{\pi}{2}\big) - \pi t^2 \sin\big(\pi x + \frac{\pi}{2}\big)$. The exact solution of this problem is $u(x,t) = t^2\cos\big(\pi x + \frac{\pi}{2}\big)$. We have solved this problem for both the numerical schemes $NS1$ and $NS2$ and obtained results are shown through Table 2.7 and Table 2.8. Fig. 2.2 shows a comparison between the numerical solution and the analytical solution for $\alpha = 0.5$ and step size $h = \tau = 1/500$. From Table 2.7, Table 2.8, and solution plot 2.2, it can be seen that numerical solution and analytical solution agree well.

TABLE 2.7: MAE and CO of the scheme $NS1$ for Ex. 2.5.4 using different values of α and τ.

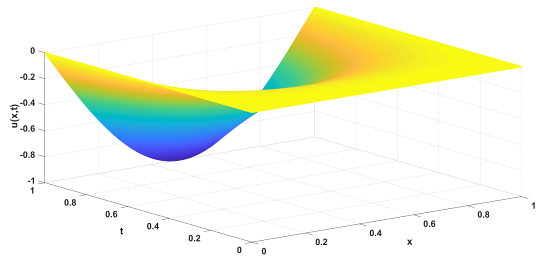
α	h	$NS1$		
		MAE	CO	CPU time (Sec.)
0.2	1/100	7.87781E-05		1.759946
	1/200	1.99191E-05	1.983644810	7.853415
	1/400	5.04418E-06	1.981456693	32.599035
	1/800	1.27951E-06	1.979031197	139.467982
	1/1200	5.74019E-07	1.976912496	419.683619
0.4	1/100	9.56176E-05		1.594386
	1/200	2.58242E-05	1.888549950	7.604817
	1/400	7.08701E-06	1.865476194	32.287809
	1/800	1.97941E-06	1.840104657	132.871470
	1/1200	9.46749E-07	1.818953614	368.463217
0.6	1/100	1.60869E-04		1.788234
	1/200	5.18199E-05	1.634312219	8.130293
	1/400	1.73349E-05	1.579824024	33.335089
	1/800	5.98610E-06	1.533995797	139.567519
	1/1200	3.25290E-06	1.504179636	398.948424

Table 2.8: MAE and CO of the scheme $NS2$ for Ex. 2.5.4 using different values of α and τ.

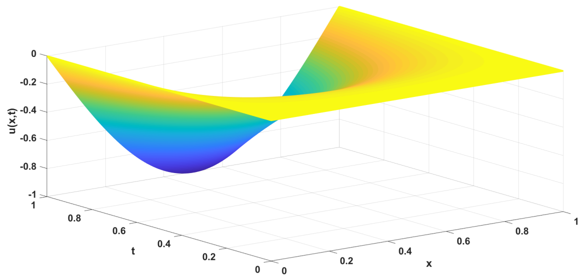
α	h	$NS2$		
		MAE	CO	CPU time (Sec.)
0.2	1/100	7.39817E-05		4.475983
	1/200	1.84950E-05	2.000031329	24.116395
	1/400	4.62383E-06	1.999976026	287.864813
	1/800	1.15596E-06	1.999994309	4627.437339
	1/1200	5.13778E-07	1.999920697	99827.576201
0.4	1/100	7.23251E-05		4.184334
	1/200	1.80824E-05	1.999909706	23.594654
	1/400	4.52072E-06	1.999963188	424.438936
	1/800	1.13022E-06	1.999951787	7764.674037
	1/1200	5.02365E-07	1.999770321	12874.887152
0.6	1/100	7.05567E-05		6.536067
	1/200	1.76427E-05	1.999716142	31.238602
	1/400	4.41095E-06	1.999906933	580.655956
	1/800	1.10279E-06	1.999925968	7629.573928
	1/1200	4.90114E-07	2.000083865	15982.439813



(a) Numerical solution using $NS1$



(b) Numerical solution using $NS2$



(c) Analytical solution

FIGURE 2.2: Comparison between numerical and analytical solution for Ex. 2.5.4 for $\alpha = 0.5$.

Remark 2.5.1. In the above-defined time fractional advection–diffusion equation (2.19), if we take $V = 0$, it reduces to a time fractional diffusion equation. By considering an example, we show that the discussed schemes $NS1$ and $NS2$ work well for the time fractional diffusion equation also which is defined in the Caputo–Prabhakar sense.

Example 2.5.5. Consider the fractional diffusion equation [99],

$$ {}_{0}^{\mathrm{CP}}D^{\gamma}_{\rho,\alpha,\omega}u(x,t) = \frac{\partial^2 u(x,t)}{\partial x^2} + F(x,t), \; 0 < x < 1, 0 < t < 1, $$

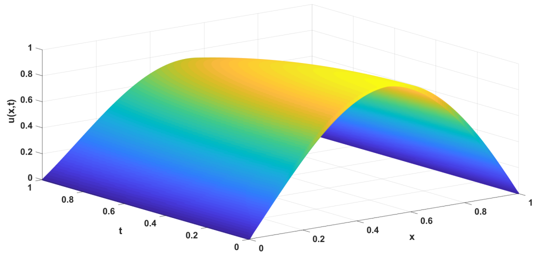
with the initial condition $u(x,0) = \sin(\pi x)$ and boundary conditions $u(0,t) = u(1,t) = 0$ and $F(x,t) = 2x(x-1)t^{2-\alpha}E^{-\gamma}_{\rho,3-\alpha}(\omega t^{\rho}) + \pi^2 \sin(\pi x) - 2t^2$. The analytical solution of this problem is $u(x,t) = \sin(\pi x) + x(x-1)t^2$. We examine MAE and CO for different values of τ and fixing $h = 1/512$, and again for different values of h and fixing $\tau = 1/600$, with $\alpha = 0.85$. The obtained numerical results are shown in Table 2.9 and Table 2.10. From Table 2.9 and Table 2.10, it is clear that error decreases as we reduce the step size which implies that numerical solutions are co-incident with the analytical one. The solution plot of the analytical and numerical solutions is also given in Fig. 2.3, which draws the same conclusion.

TABLE 2.9: MAE and CO of the schemes $NS1$ and $NS2$ for Ex. 2.5.5 for different values of τ and fixing $h = 1/512$ when $\alpha = 0.85$.

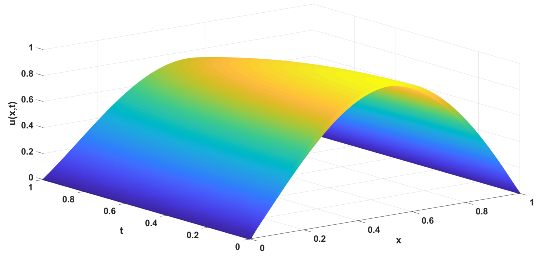
τ	$NS1$		$NS2$	
	MAE	CO	MAE	CO
1/10	1.67889E-03		4.78338E-04	
1/20	7.50178E-04	1.1622106712	1.38449E-04	1.788674320
1/40	3.33834E-04	1.1681013956	3.80566E-05	1.863134649
1/80	1.47578E-04	1.1776528864	1.06408E-05	1.838532264
1/160	6.43087E-05	1.1983931954	3.10503E-06	1.776936275

TABLE 2.10: MAE and CO of the schemes $NS1$ and $NS2$ for Ex. 2.5.5 using different values of h and fixing $\tau = 1/600$ when $\alpha = 0.85$.

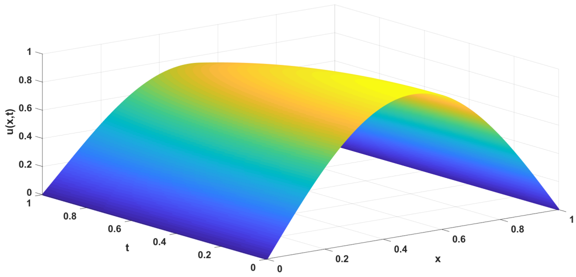
h	$NS1$		$NS2$	
	MAE	CO	MAE	CO
1/10	8.00858E-03		8.02309E-03	
1/20	1.98427E-03	2.012935427	1.99880E-03	2.005024051
1/40	4.84736E-04	2.033339791	4.99266E-04	2.001254023
1/80	1.10259E-04	2.136303870	1.24789E-04	2.000310668
1/160	1.66651E-05	2.725994872	3.11959E-05	2.000066621



(a) Numerical solution using $NS1$



(b) Numerical solution using $NS2$



(c) Analytical solution

FIGURE 2.3: Comparison between numerical and analytical solution for Ex. 2.5.5 for $\alpha = 0.5$.

2.6 Conclusions

In this chapter, we discussed two numerical schemes namely, $NS1$ and $NS2$, to approximate the Caputo–Prabhakar derivative. Convergence order for the schemes $NS1$ and $NS2$ comes out to be $(2 - \alpha)$ and $(3 - \alpha)$, respectively. The properties of the discretization coefficients and truncation errors of the schemes are discussed. Further, we applied these schemes for solving the time fractional advection–diffusion equation numerically. We also discussed the stability analysis of the schemes. From the numerical examples, it is clear that numerical results validate our analytical findings. We also discussed the time fractional diffusion equation defined in the Caputo–Prabhakar sense in Ex. 2.5.5. Observing the results, we can say that the schemes also work well for the fractional diffusion equation. In future, these schemes can be applied to solve many other physical phenomena defined in the Caputo–Prabhakar sense. To increase computational efficiency, we will try to construct schemes of a much higher convergence order in the future.

****