JAM CM Journal of Applied Mathematics and Computational Mechanics 2021, 20(2), 5-16
www.amcm.pcz.pl p-ISSN 2299-9965
DOI: 10.17512/jamcm.2021.2.01 e-ISSN 2353-0588

A NEW NUMERICAL SCHEME FOR SOLVING THE TWO DIMENSIONAL FRACTIONAL DIFFUSION EQUATION

Dilara Altan Koç, Mustafa Gülsu

Department Mathematics, Mugla Sitki Kocman University, Turkey
dilaraaltan@mu.edu.tr, mgulsu @mu.edu.tr

Received: 2 May 2021; Accepted: 23 June 2021

Abstract. In this study, the locally one dimensional (LOD) method is used to solve the two dimensional time fractional diffusion equation. The fractional derivative is the Caputo fractional derivative of order α. The rate of convergence of the finite difference method is presented. It is seen that this method is in agreement with the obtained numerical solutions with acceptable central processing unit time (CPU time). Error estimates, numerical and exact results are tabulated. The graphics of errors are given.

MSC 2010: *65M06, 65M22, 34K37*
Keywords: *finite difference method, fractional calculus, numerical methods*

1. Introduction

The diffusion equation with fractional order are used to model problems in physics [1], finance [2], hydrology [3] and anomalous equations [4]. Anomalous diffusion has an important role in the literature to describe many physical events. The time fractional diffusion equation is obtained from the classical diffusion equation by replacing the first order time derivative by a fractional derivative of order α, in the Caputo, Grünwald or Riemann Liouville sense.

Some of the common numerical methods [5] for solving fractional diffferential equations, such as the variational iteration method [6], Green's function solution [7] and the Mittag Leffler function solution [8].

There are many numerical methods used to solve the fractional differential equation. P. Zhuang and F. Liu used an implicit difference approximation method [9], C. Tadjeran and M.M. Meerschaert presented a numerical method to solve a fractional superdiffusive differential equation. In this method, the alternating directions implicit (ADI) approach, the Crank-Nicolson discretization and the Richardson extrapolation are used [10]. In [11] the Grünwald finite difference approximation is used for the fractional derivative with ADI method and locally one dimensional (LOD) method is used in [12-16].

In this study, we consider the two dimensional time fractional diffusion equation of the form [9]

$$\frac{\partial^\alpha u(x,y,t)}{\partial t^\alpha} = A(x,y,t)\frac{\partial^2 u(x,y,t)}{\partial x^2} + B(x,y,t)\frac{\partial^2 u(x,y,t)}{\partial y^2} + f(x,y,t) \quad (1)$$

with the following initial and boundary conditions:

$$u(x,y,0) = \varphi(x,y), \ \ (x,y) \in \Omega \quad (2)$$

$$u(x,y,t)|_{\partial\Omega} = 0, \ \ 0 \le t \le M \quad (3)$$

where $\Omega = \{(\mathrm{x},\mathrm{y})|0 \le \mathrm{x} \le \mathrm{L}_1, 0 \le \mathrm{y} \le \mathrm{L}_2\}$, $A(x,y,t) > 0$, $B(x,y,t) > 0$ and $f(x,y,t)$ is a suitable function.

Equation (1) is solved by the LOD method. The stability of the method is discussed using mathematical induction, and it is shown to be unconditionally stable. To obtain the discrete form of Eq. (1), the central difference with $O((dx)^2)$ and $O((dy)^2)$ for the second order derivative in x and y and the Caputo fractional derivative definition for the fractional derivative in t are used.

The rest of this paper is organizated as follows. The numerical method used is described in Section 2. Stability analysis is mentioned in Section 3. Numerical experiments and numerical results are given in Section 4. Finally, the main results are summarized in Section 5.

2. Numerical method

To solve two-dimensional differential equation, one of the finite difference methods which is the locally one-dimensional method is considered. Discrete grids with uniform steps are defined as follows:

$$\omega_{dx} = \{x_i = idx, \ i = 1,2,\dots,N_1 - 1, \ N_1 dx = L_1\}, \quad (4)$$

$$\bar{\omega}_{dx} = \omega_{dx} \cup \{x_0 = 0, \ x_{N_1} = L_1\}, \quad (5)$$

$$\omega_{dy} = \{y_j = jdy, \ j = 1,2,\dots,N_2 - 1, \ N_2 dy = L_2\}, \quad (6)$$

$$\bar{\omega}_{dy} = \omega_{dy} \cup \{y_0 = 0, \ y_{N_2} = L_2\}, \quad (7)$$

$$\bar{\omega} = \bar{\omega}_{dx} \times \bar{\omega}_{dy}, \quad (8)$$

$$\omega_{dt} = \{t_k = kdt, k = 1,2,\dots,M, Mdt = T\}, \ \bar{\omega}_{dt} = \omega_{dt} \cup \{t_0 = 0\}. \quad (9)$$

The notation $u_{i,j}^k = u(x_i, y_j, t_k)$ is for functions defined on $\omega \times \bar{\omega}$ and the notation $u_{i,j}^{k+1/2} = u(x_i, y_j, t_k + 0.5dt)$.

The symmetric second difference quotient in space at level t_k for approximating the second order space derivatives with $O((dx)^2)$ and $O((dy)^2)$ are defined.

$$\frac{\partial^2 u_{i,j}^k}{\partial x^2} = \frac{u_{i-1,j}^k - 2u_{i,j}^k + u_{i+1,j}^k}{(dx)^2}, \quad \frac{\partial^2 u_{i,j}^k}{\partial y^2} = \frac{u_{i,j-1}^k - 2u_{i,j}^k + u_{i,j+1}^k}{(dy)^2} \tag{10}$$

The time fractional derivative term can be approximated by the following scheme [9]:

$$\begin{aligned}\frac{\partial^\alpha u_{i,j}^k}{\partial t^\alpha} &= \frac{(dt)^{-\alpha}}{\Gamma(2-\alpha)}\left[u_{i,j}^k - u_{i,j}^{k-1}\right] \\ &\quad + \frac{(dt)^{-\alpha}}{\Gamma(2-\alpha)} \sum_{s=1}^{k-1} \left[u_{i,j}^{k-s} - u_{i,j}^{k-s-1}\right]\left[(s+1)^{1-\alpha} - s^{1-\alpha}\right]\end{aligned} \tag{11}$$

We accomplish the transition from the kth level of time to the $(k+1)$th level by splitting it into two stages, and one dimensional finite difference subproblems in each of them are solved.

If a LOD procedure is applied, Eq. (1) is split into the following one dimensional equations:

$$\frac{1}{2}\frac{d^\alpha u}{dt^\alpha} = A(x,y,t)\frac{d^2 u}{dx^2} + \frac{1}{2} f(x,y,t) \tag{12}$$

$$\frac{1}{2}\frac{d^\alpha u}{dt^\alpha} = B(x,y,t)\frac{d^2 u}{dy^2} + \frac{1}{2} f(x,y,t) \tag{13}$$

To obtain the solution from t^k to t^{k+1}, we assume that Eq. (12) holds from t^k to $t^{k+1/2}$ and Eq. (13) holds from $t^{k+1/2}$ to t^{k+1}.

The first subproblem is implicit with respect to x,

$$\begin{aligned}&-Q_1 u_{i-1,j}^{k+1/2} + (1+2Q_1)u_{i,j}^{k+1/2} - Q_1 u_{i+1,j}^{k+1/2} \\ &\qquad = u_{i,j}^k - \sum_{s=1}^{k} \left(u_{i,j}^{k-s+1/2} - u_{i,j}^{k-s}\right)\left((s+1)^{1-\alpha} - s^{1-\alpha}\right) \\ &\qquad + \Gamma(2-\alpha)(dt)^\alpha f_{i,j}^k\end{aligned} \tag{14}$$

The second subproblem is also implicit with respect to y,

$$\begin{aligned}&-Q_2 u_{i,j-1}^{k+1} + (1+2Q_2)u_{i,j}^{k+1} - Q_2 u_{i,j+1}^{k+1} \\ &\qquad = u_{i,j}^{k+1/2} - \sum_{s=1}^{k} \left(u_{i,j}^{k-s+1} - u_{i,j}^{k-s+1/2}\right)\left((s+1)^{1-\alpha} - s^{1-\alpha}\right) \\ &\qquad + \Gamma(2-\alpha)(dt)^\alpha f_{i,j}^k\end{aligned} \tag{15}$$

By using the matrix representations of systems Eq. (14) and Eq. (15), on each fixed point $y = y_j$, $j = 1,..,N_2 - 1$ a set of $N_1 - 1$ equations at the points x_i is solved and the solutions $u_{i,j}^{n+1/2}$ are obtained. Then, in the second step, a set of $N_2 - 1$ equations at the points y_j is solved to obtain $u_{i,j}^{n+1}$. The LOD method procedure is stated as the following.

Algorithm

Step 1.
Input the time step (dt), and space steps (dx) and (dy).
Step 2.
Compute grid points "x_i" for $i = 0, \dots, N_1 - 1$, "y_j" for $j = 0, \dots, N_2 - 1$ and "t_k" for $k = 0, \dots, M - 1$.
Step 3.
Calculate the given initial conditions $u_{i,j}^0, (x_i, y_j) \in \overline{\omega}$ for $i = 0, \dots, N_1 - 1$, $j = 0, \dots, N_2 - 1$.
Step 4.
Construct the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ for system (14).
Step 5.
Compute $\boldsymbol{A}$, $\boldsymbol{B}$ and for each $x_i \in \omega_{dx}$, solve system (14) and calculate $u_{i,j}^{k+1/2}$, $y_j \in \omega_{dy}$.
Step 6.
Construct the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ for system (15).
Step 7.
Compute $\boldsymbol{A}$, $\boldsymbol{B}$ and for each $y_j \in \omega_{dy}$, solve system (15) and calculate $u_{i,j}^{k+1}$, $x_i \in \omega_{dx}$.
Step 8.
Evaluate absolute errors.

Let $u_{i,j}^k$ be the approximations of $U_{i,j}^k$, and define the error function as follows:

$$error(h) = \max_{i,j} \left| u_{i,j}^M - U_{i,j}^M \right|, \text{ for } k = 1,2, \dots, M.$$

The rate of convergence of the method used are estimated through the asymptotic formula [14],

$$rate = \frac{log\left(error(h_1)/error(h_2)\right)}{log(h_1/h_2)}$$

in which $error(h_1)$ and $error(h_2)$ are absolute errors based on different mesh sizes $h = h_1$ and $h = h_2$.

3. Stability analysis

The stability of the method will be looked at separately for each step. Eq. (11) is written in Eq. (12) and Eq. (13) instead of the fractional derivative and for the second order derivative, Eq. (10) is used. Then we have [9],

$$\begin{aligned}-\mu_1\left(u_{i+1,j}^{k+1/2}-u_{i-1,j}^{k+1/2}\right)+(1+2\mu_1)u_{i,j}^{k+1/2}\\ =u_{i,j}^{k}-\sum_{s=1}^{k}u_{i,j}^{k-s+1/2}b_s+\sum_{s=1}^{k}u_{i,j}^{k-s}b_s\\ +\Gamma(2-\alpha)(dt)^{\alpha}f_{i,j}^{k+1/2}\end{aligned} \tag{16}$$

where

$$\frac{\Gamma(2-\alpha)(dt)^{\alpha}Q_1}{(\Delta x)^2}=\mu_1,\ \ (s+1)^{1-\alpha}-s^{1-\alpha}=b_s$$

According to the states of k, the following two equations are obtained:

$$\begin{aligned}&k=0\\ &-\mu_1\left(u_{i+1,j}^{1/2}-u_{i-1,j}^{1/2}\right)+(1+2\mu_1)u_{i,j}^{1/2}=u_{i,j}^{0}+\Gamma(2-\alpha)(dt)^{\alpha}f_{i,j}^{1/2}\end{aligned} \tag{17}$$

$$\begin{aligned}&k>0\\ &-\mu_1\left(u_{i+1,j}^{k+1/2}-u_{i-1,j}^{k+1/2}\right)+(1+2\mu_1)u_{i,j}^{k+1/2}\\ &=(2-2^{1-\alpha})u_{i,j}^{k}+\sum_{s=1}^{k-1}u_{i,j}^{k-s+1/2}(b_{s-1}-b_s)+b_k u_{i,j}^{0}\\ &+\Gamma(2-\alpha)(dt)^{\alpha}f_{i,j}^{k+1/2}\end{aligned} \tag{18}$$

where $i=0,1,\dots,l,\ \ j=0,1,\dots,m,\ \ k=0,1,\dots,n$ and let $\tilde{u}\,_{i,j}^{k}$ be the approximate solution, and the term of error is $\varepsilon_{i,j}^{k}=\tilde{u}\,_{i,j}^{k}-u_{i,j}^{k}$. If this error term is written in Eq. (17) and Eq. (18), the following equations are obtained:

$$k=0\quad -\mu_1\left(\varepsilon_{i+1,j}^{1/2}-\varepsilon_{i-1,j}^{1/2}\right)+(1+2\mu_1)\varepsilon_{i,j}^{1/2}=\varepsilon_{i,j}^{0} \tag{19}$$

$$\begin{aligned}k>0\quad -\mu_1\left(\varepsilon_{i+1,j}^{k+1/2}-\varepsilon_{i-1,j}^{k+1/2}\right)+(1+2\mu_1)\varepsilon_{i,j}^{k+1/2}\\ =(2-2^{1-\alpha})\varepsilon_{i,j}^{k}+\sum_{s=1}^{k-1}\varepsilon_{i,j}^{k-s+1/2}(b_{s-1}-b_s)+b_k\varepsilon_{i,j}^{0}\end{aligned} \tag{20}$$

which can be written as

$$\begin{aligned}&\boldsymbol{A}\boldsymbol{E}^{1}=\boldsymbol{E}^{0}\\ &\boldsymbol{E}^{k+1}=(b_0-b_1)\boldsymbol{E}^{k}+(b_1-b_2)\boldsymbol{E}^{k-1}+\cdots+(b_{k-1}-b_k)\boldsymbol{E}^{1}+b_k\boldsymbol{E}^{0}\\ &\boldsymbol{E}^{0}\end{aligned}$$

where

$$\boldsymbol{E}_i^k = \begin{bmatrix} \varepsilon_{i,1}^k & \varepsilon_{i,2}^k & \cdots & \varepsilon_{i,m-1}^k \end{bmatrix}^T \quad i = 0,1,\ldots,l$$

and

$$\boldsymbol{E}^k = \begin{bmatrix} \boldsymbol{E}_1^k & \boldsymbol{E}_2^k & \cdots & \boldsymbol{E}_{l-1}^k \end{bmatrix}^T$$

Hence, the following result can be proved using mathematical induction [9].

Theorem 1. $\left\|\boldsymbol{E}^{k+1/2}\right\|_\infty \leq \|\boldsymbol{E}^0\|_\infty,\ k = 0,1,2,3,\ldots..$

Proof. For $k = 0$

$$-\mu_1\left(\varepsilon_{i+1,j}^{1/2} - \varepsilon_{i-1,j}^{1/2}\right) + (1 + 2\mu_1)\varepsilon_{i,j}^{1/2} = \varepsilon_{i,j}^0$$

$$\left|\varepsilon_{p,q}^{1/2}\right| = \max_{1\leq i\leq l-1,\ 1\leq j\leq m-1} \left|\varepsilon_{i,j}^{1/2}\right|$$

$$\begin{aligned}
\left|\varepsilon_{p,q}^{1/2}\right| &= -\mu_1\left(\left|\varepsilon_{p,q}^{1/2}\right| + \left|\varepsilon_{p,q}^{1/2}\right|\right) + (1 + 2\mu_1)\left|\varepsilon_{p,q}^{1/2}\right| \\
&\leq -\mu_1\left(\left|\varepsilon_{p+1,q}^{1/2}\right| + \left|\varepsilon_{p-1,q}^{1/2}\right|\right) + (1 + 2\mu_1)\left|\varepsilon_{p,q}^{1/2}\right| \\
&\leq \left|-\mu_1\left(\varepsilon_{p+1,q}^{1/2} + \varepsilon_{p-1,q}^{1/2}\right) + (1 + 2\mu_1)\varepsilon_{p,q}^{1/2}\right| = \left|\varepsilon_{p,q}^0\right| \leq \|\boldsymbol{E}^0\|_\infty
\end{aligned}$$

then $\left\|\boldsymbol{E}^{1/2}\right\|_\infty \leq \|\boldsymbol{E}^0\|_\infty$ is held.

Suppose that for $s = 0,1,2,\ldots,k,\ \left\|\boldsymbol{E}^{s+1/2}\right\|_\infty \leq \|\boldsymbol{E}^0\|_\infty$.

Let $\left|\varepsilon_{p,q}^{(k+1/2)+1}\right| = \max_{1\leq i\leq l-1,\ 1\leq j\leq m-1} \left|\varepsilon_{i,j}^{(k+1/2)+1}\right|$

we have,

$$\begin{aligned}
\left|\varepsilon_{p,q}^{(k+1/2)+1}\right| &= -\mu_1\left(\left|\varepsilon_{p,q}^{(k+1/2)+1}\right| + \left|\varepsilon_{p,q}^{(k+1/2)+1}\right|\right) + (1 + 2\mu_1)\left|\varepsilon_{p,q}^{(k+1/2)+1}\right| \\
&\leq -\mu_1\left(\left|\varepsilon_{p+1,q}^{(k+1/2)+1}\right| + \left|\varepsilon_{p-1,q}^{(k+1/2)+1}\right|\right) + (1 + 2\mu_1)\left|\varepsilon_{p,q}^{(k+1/2)+1}\right| \\
&= \left|(b_0 - b_1)\varepsilon_{p,q}^{k+1/2} + \sum_{s=1}^{k-1} \varepsilon_{p,q}^{k-s}(b_s - b_{s+1}) + b_k\varepsilon_{p,q}^0\right| \\
&\leq (b_0 - b_1)\left|\varepsilon_{p,q}^{k+1/2}\right| \\
&\quad + \sum_{s=1}^{k-1}\left|\varepsilon_{p,q}^{k-s}\right|(b_s - b_{s+1}) + b_k\left|\varepsilon_{p,q}^0\right| \\
&\leq (b_0 - b_1)\left\|\boldsymbol{E}^{k+1/2}\right\|_\infty \\
&\quad + \sum_{s=1}^{k-1}\left\|\boldsymbol{E}^{k-s}\right\|_\infty(b_s - b_{s+1}) + b_k\|\boldsymbol{E}^0\|_\infty \\
&\leq \left(b_0 - b_1 + \sum_{s=1}^{k-1}(b_s - b_{s+1}) + b_k\right)\|\boldsymbol{E}^0\|_\infty = \|\boldsymbol{E}^0\|_\infty
\end{aligned}$$

also $\left\|\boldsymbol{E}^{k+1/2}\right\|_\infty \leq \|\boldsymbol{E}^0\|_\infty$.

If the same operations are applied for Eq. (15), it is seen to be stable for the other step, which is from $(k+1/2)$ step to $(k+1)$.

The LOD method approximation defined by (14) and (15) is unconditionally stable.

4. Numerical experiments

In this section, in order to demonstrate the efficiency of the method, we solved two problems whose exact solutions are given.

Problem 1.

Consider the two-dimensional time fractional diffusion problem Eq. (1) subject to the following conditions [9],

$$u(x,y,t)|_{\partial\Omega} = 0,\ \ u(x,y,0) = sin\pi x sin\pi y \tag{21}$$

$$\Omega = \{(x,y)|0 < x < 1, 0 < y < 1\}$$

$$f(x,y,t) = \frac{25t^{(2-\alpha)}}{12\Gamma(1-\alpha)}(t^2+2)sin\pi x sin\pi y \tag{22}$$

$$A(x,y,t) = \frac{2t^{(2-\alpha)}}{\pi^2\Gamma(1-\alpha)},\ \ B(x,y,t) = \frac{t^{(2-\alpha)}}{12\pi^2\Gamma(1-\alpha)} \tag{23}$$

For the problem (1) with conditions (21)-(23), the coefficients Q_1 and Q_2 written as the following:

$$Q_1 = \frac{2t_n{}^{1,6}\Gamma(2-\alpha)(dt)^\alpha}{(dx)^2\pi^2\Gamma(0,6)},\ \ Q_2 = \frac{t_n{}^{1,6}\Gamma(2-\alpha)(dt)^\alpha}{(dx)^2 12\pi^2\Gamma(0,6)}$$

The exact solution is of (1) with (21)-(23) conditions, $u(x,y,t) = (t^2 + 1)sin\pi x sin\pi y$.

This section has tables including numerical, exact solution and errors in different mesh point for different orders. Take $dx = dy = h$.

In Table 1, the results are shown for $u_{i,15}^{20}$ with $i = 0, \dots, 10$, the time step is $t = 0.025$ and the order of fractional derivative is $\alpha = 0.4$. In Table 2, the rate of convergence and the CPU time for different h are given. In Table 3, the results are shown for $u_{i,15}^{20}$ with the same values as in Table 1 and the different order.

Table 1. Numerical-exact solutions and errors for Eq. (1), (21)-(23) with $h = 0.05$, $dt = 0.00125$, $\alpha = 0.4$, $t = 0.025$

x	Numerical solution	Exact solution	Error
0.0	0.11056998	0.11068500	0.00011501
0.05	0.21841737	0.21864457	0.00022720
0.1	0.32088660	0.32122039	0.00033379
0.15	0.41545453	0.41588670	0.00043216
0.20	0.49979260	0.50031250	0.00051989
0.25	0.57182411	0.57241894	0.00059482
0.30	0.62977541	0.63043052	0.00065511
0.35	0.67221956	0.67291882	0.00069926

Table 2. The rate of convergence and the CPU time for different h and $T_{end} = 1$

h	Error	Rate	CPU time
1/10	0.03384810	–	15 sec
1/20	0.00868640	1.96	70 sec
1/30	0.00387932	1.94	220 sec
1/40	0.00218581	2.06	300 sec

Table 3. Numerical-exact solutions and errors for Eq. (1), (21)-(23) with $h = 0.05$, $dt = 0.00125$, $\alpha = 0.0004$, $t = 0.025$

x	Numerical solution	Exact solution	Error
0.0	0.11059861	0.11068500	0.00008638
0.05	0.21847393	0.21864457	0.00017064
0.1	0.32096969	0.32122039	0.00025070
0.15	0.41556211	0.41588670	0.00032458
0.20	0.49992202	0.50031250	0.00039047
0.25	0.57197218	0.57241894	0.00044675
0.30	0.62993849	0.63043052	0.00049203
0.35	0.67239363	0.67291882	0.00052519

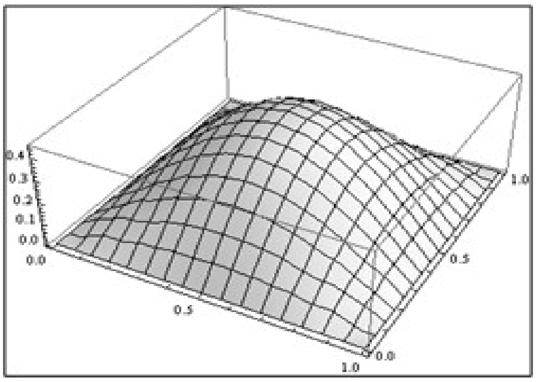

Fig. 1. 3D graph of absolute error in Problem 1 for $h = \frac{1}{10}$, $0 \leq x \leq 1, 0 \leq y \leq 1$ at $t = 1$

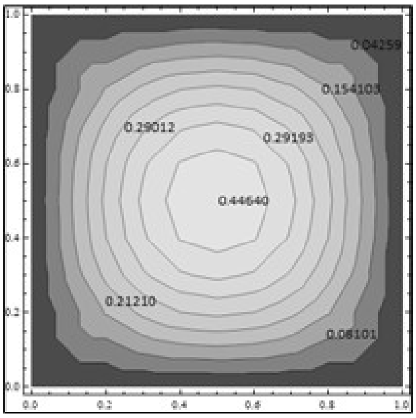

Fig. 2. Contour graph of absolute error in Problem 1 for $h = \frac{1}{10}$

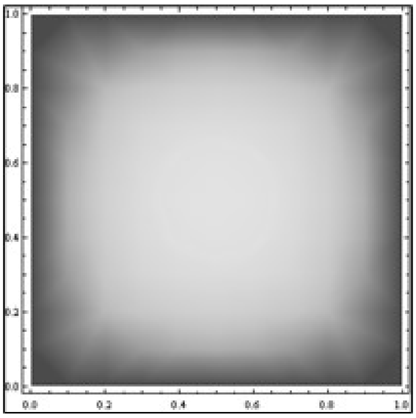

Fig. 3. Density graph of absolute error in Problem 1 for $h = \frac{1}{10}$

Problem 2.

Consider the two-dimensional diffusion problem (1) subject to the initial condition [15]

$$u(x, y, 0) = 0 \tag{24}$$

and the boundary condition

$$u(x, y, t)|_{\partial\Omega} = 0, \tag{25}$$

$\Omega = \{(x, y)|0 < x < \pi, 0 < y < \pi\}$ and $\partial\Omega$ is the boundary of Ω.

$$f(x, y, t) = \left(\frac{\Gamma(\alpha + 3)}{2} t^2 + 2t^{\alpha+2}\right) sinxsiny \tag{26}$$

$$A(x, y, t) = B(x, y, t) = 1 \tag{27}$$

For the problem (1) with conditions (24)-(27), the coefficients Q_1 and Q_2 written as the following

$$Q_1 = Q_2 = \frac{\Gamma(2-\alpha)(dt)^{\alpha}}{(dx)^2},$$

The exact solution is of (1) with (24)-(27) conditions, $u(x,y,t) = (t^{\alpha+3})sinxsiny$.

In Table 4, numerical-exact solutions and errors are given for $\alpha = 0.1$ and $x = y = 1{,}41$, different time steps for $dt = 0.001$. The rate of convergence and the CPU time for a different h are given in Table 5.

Table 4. Numerical-exact solutions and errors for Eq. (1), (19)-(22) with $h = 0.05$, $dt = 0.001$

x	Numerical solution	Exact solution	Error
0.0	0.371172e-6	0.735880e-7	0.297584e-6
0.01	0.493086e-7	0.630957e-6	0.581648e-6
0.02	0.151825e-7	0.221759e-5	0.220241e-5
0.03	0.973888e-8	0.540994e-5	0.540020e-5
0.04	0.685596e-8	0.108047e-4	0.107978e-4
0.05	0.571549e-8	0.190141e-4	0.190083e-4
0.06	0.453485e-8	0.306627e-4	0.306582e-4
0.07	0.406969e-8	0.463858e-4	0.463818e-4
0.08	0.339779e-8	0.668280e-4	0.668246e-4

Table 5. The rate of convergence and the CPU time for different h and $T_{end} = 1$

h	Error	Rate	CPU time
π/10	2.947734e-19	–	17 sec
π/20	2.268427e-18	2.94	70 sec
π/30	0.782234e-18	2.62	180 sec

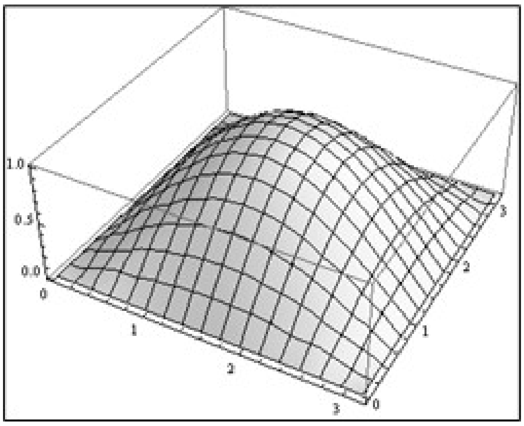

Fig. 4. 3D graph of absolute error in Problem 2 for $h = \pi/10$, $0 \le x, y \le \pi$ at $t = 1$

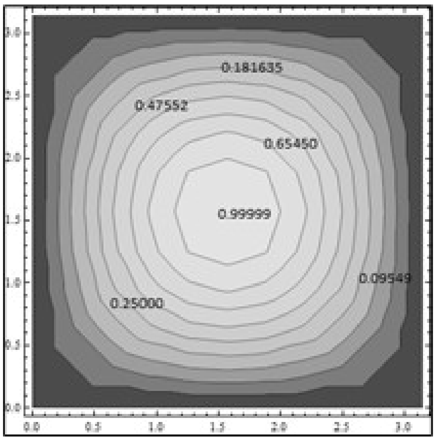

Fig. 5. Contour graph of absolute error in Problem 2 for $h = \pi/10$ $0 \leq x, y \leq \pi$ at $t = 1$

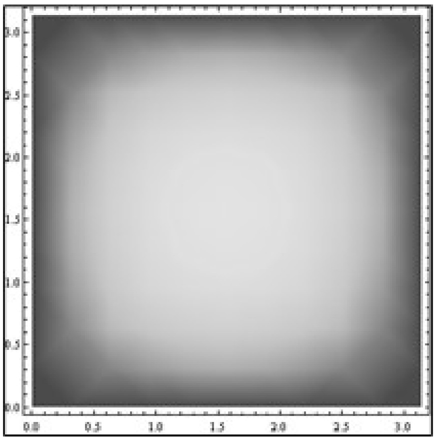

Fig. 6. Density graph of absolute error in Problem 2 for $h = \pi/10$ $0 \leq x, y \leq \pi$ at $t = 1$

5. Conclusions

In this paper, splitting difference schemes for solving the two dimensional time fractional diffusion equation is proposed. Based on the LOD strategy, we seperate the two-dimensional equation into the two one-dimensional equation. These equations form two implicit systems. After solving these systems, the results are compared with the exact solution and are supported them with tables and graphs. The convergent rate is consistent with the order of the cutting error in the finite differences and numerical experiments, and the CPU time show that this scheme effective and acceptable. This method seems particularly suitable in the sense that it can be easily applied to fractional diffusion equations. With the LOD method process, more discrete numerical solution values are reached with fewer errors.

The method and techniques proposed in this paper can also be applied to solve different kinds of fractional partial differential equations and also fractional partial delay differential equations.

References

[1] Metzler, R., & Klafter, J. (2004). The restaurant at the end of the random walk: recent developments in the description of anomalous transport by fractional Dynamics. *Journal of Physics A*, *37*, R161-R208.

[2] Chen, T., & Wang, D. (2020). Combined application of blockchain technology in fractional calculus model of supply chain financial system. *Chaos, Solitons & Fractals*, *131*, 109461.

[3] Benson, D.A., Meerschaert, M.M., & Revielle, J. (2013). Fractional calculus in hydrologic modelling: a numerical perspective. *Advances in Water Resources*, *51*, 479-497.

[4] Luchko, Y. (2016). A new fractional calculus model for the two-dimensional anomalous diffusion and its analysis. *Mathematical Modelling of Natura Phenımena*, *11*-3, 1-17.

[5] Li, Z., Liu, L., Dehghan, S., Chen, Y., & Xue, D. (2017). A review and evaluation of numerical tools for fractional calculus and fractional order controls. *International Journal of Controls*, *90*(6), 1165-1181.

[6] Wu, G.C., & Baleanu D. (2013). Variational iteration method for fractional calculus – a universal approach by Laplace transform. *Advances in Difference Equations*, *18*, 1-9.

[7] Langlands, T. (2006). Fractional Dynamics. *Physica A*, *367*, 136.

[8] Narahari, B.A., & Hanneken J. (2004). Fractional radial diffusion in a cylinder. *Journal of Molecular Liquids*, *114*, 147-151.

[9] Zhuang, P., & Liu, F. (2007). Finite difference approximation for two-dimensional time fractional diffusion equation. *Journal of Algorithms & Computational Technology*, *1*, 1.

[10] Tadjeran, C., & Meerschaert, M.M. (2007). A second order accurate numerical method for the two-dimensional fractional diffusion equation. *Journal of Computational Physics*, *220*, 813-823.

[11] Meerschaert, M.M., Scheffler, H.P., & Tadjeran C. (2006). Finite difference methods for two-dimensional fractional dispersion equation. *Journal of Computational Physics*, *211*, 249-261.

[12] Dehghan, M. (1999). Implicit locally one-dimensional methods for two-dimensional diffusion with a non-local boundary condition. *Mathematics and Computers in Simulation*, *49*, 331-349.

[13] Chen, J., & Ge, Y. (2018). High order locally one dimensional methods for solving two-dimensional parabolic equation. *Advence in Difference Equations*, 361.

[14] Soori, Z., & Aminataei, A. (2018). Effect of the nodes near boundary points on the stability analysis of sixth-order compact finite difference ADI scheme for the two-dimensional time fractional diffusion-wave equation. *Transactions of A. Razmadze Mathematical Institute*, *172*, 582-605.

[15] Kutluay, S., & Yağmurlu, N.M. (2012). The modified bi-quintic b-splines for solving the two dimensional unsteady burgers' equation. *European International Journal of Science and Technology*, *1*, 2.

[16] Kutluay, S., & Yağmurlu, N.M. (2013). Derivation of the modified bi-quintic b-spline base functions: an application to Poisson equation. *American Journal of Computational and Applied Mathematics*, *3*(1), 26-32.