Fractional Newton Explicit Group Method for Time-Fractional Nonlinear Porous Medium Equations

Article · July 2024

DOI: 10.18576/pfda/100305

4 authors:

Jackel Vui Lung Chew
Universiti of Malaysia Sabah

Andang Sunarto
Universitas Islam Negeri Fatmawati Sukarno Bengkulu

Jumat Sulaiman
Universiti of Malaysia Sabah

Mohammad Fadhli Asli
Universiti of Malaysia Sabah

Progress in Fractional Differentiation and Applications
An International Journal

http://dx.doi.org/10.18576/pfda/100305

Fractional Newton Explicit Group Method for Time-Fractional Nonlinear Porous Medium Equations

Jackel Vui Lung Chew[1,*], *Andang Sunarto*[2], *Jumat Sulaiman*[3] *and Mohammad Fadhli Asli*[1]

[1] Faculty of Computing and Informatics, Universiti Malaysia Sabah Labuan International Campus, Labuan, 87000, Malaysia
[2] Tadris Matematika, Universitas Islam Negeri (UIN) Fatmawati Sukarno, Bengkulu, 38211, Indonesia
[3] Faculty of Science and Natural Resources, Universiti Malaysia Sabah, Kota Kinabalu, Sabah, 88400, Malaysia

Received: 23 Jan. 2023, Revised: 14 Oct. 2023, Accepted: 4 Dec. 2023
Published online: 1 Jul. 2024

Abstract: This paper presents a fractional Newton explicit group method to solve time-fractional nonlinear porous medium equations. The presented method utilizes implicit finite difference schemes with the Caputo time-fractional derivative operator. This paper aims to evaluate the accuracy and efficiency of the proposed method in solving initial boundary value problems of porous medium equations at different orders of time-fractional derivatives. The method is experimented repeatedly by using several large systems of equations to illustrate the consistency of the method's performance. In addition, the method is also experimented in solving some physics problems, which can show the method's efficacy in solving realistic phenomena.

Keywords: Porous medium equation, time-fractional, Caputo, finite difference method, Newton method, explicit group.

1 Introduction

Time-fractional porous medium equation (TFPME) is one of the important equations to describe the diffusion of matter subjects to a porous medium setting. For the past few years, TFPME has presented in the modelling of anomalous diffusion that occurred in the stock markets [1], wet porous medium [2], and signal processing [3]. The early applications of TFPME to mathematical describe such realistic phenomena have made it one of the interests of several researchers. Some researchers focus on studying the theory behind the solution of TFPME. For instance, Imbert et al. [4] studied the regularity of solutions of TFPME. Then, Li et al. [5] expanded the regularity of solutions of TFPME and provided proof of the well-posedness and regularity of solutions to TFPME. Next, Dao [6] studied solutions of a general fractional porous medium equation with a non-Lipschitz absorption term. His work showed weak solutions, Lp-estimates, decay estimates, and the disappearance of weak solutions after a finite time. Moreover, Wittbold et al. [7] investigated bounded weak solutions of TFPME. Recently, Yang and Wang [8] performed Lie group analysis to investigate TFPME and constructed some explicit group-invariant solutions. The theory about the solutions of TFPME is still expanding, and to date, it is sufficient to provide a fundamental understanding of the solutions of TFPME.

Apart from the group of researchers that focus on establishing the theory, some researchers developed and proposed different methods that can be applied to finding the solutions of TFPME. For instance, Plociniczak [9] proposed a method to reduce a TFPME into a Volterra integral equation and obtain self-similar solutions. Then, Liu et al. [10] suggested the Lie symmetry approach to obtain invariant solutions and converted TFPME to the fractional ordinary differential equation. Recently, Plociniczak and Plociniczak [11] extended the method by [9] to obtain self-similar solutions of TFPME subjects to Dirichlet, Neumann, and Robin boundaries on the half-line. Most of the previous work used the conversion from a TFPME to other differential equations for self-similar solutions. In addition, there is limited availability of study that utilizes the finite difference method to solve TFPME specifically. Therefore, this paper introduces an efficient iterative method based on the implicit finite difference method without converting the original TFPME to another form of differential equations.

* Corresponding author e-mail: jackelchew93@ums.edu.my

In this paper, a fractional Newton explicit group method is developed to solve TFPME. The method utilizes implicit finite difference schemes with the Caputo time-fractional derivative operator. The accuracy and efficiency of the proposed method is evaluated on different orders of time-fractional derivatives. The paper is organized as follows. Section 2 describes the numerical method and its systematic formulations. Section 3 discusses the numerical experiment and illustrates the results with discussions. The conclusion and future work are described in Section 4.

2 Numerical method

This paper aims to study the solution $s = s(x,t)$ of the following reduced form of TFPME,

$$\frac{\partial^\alpha s}{\partial t^\alpha} = \frac{\partial}{\partial x}\left(s^m \frac{\partial s}{\partial x}\right), 0 < \alpha < 1, m \in \mathbb{N}, \tag{1}$$

subject to the initial condition

$$s_0 = s(x,0), 0 \le x \le 1, \tag{2}$$

and boundary conditions

$$s_a = s(a,t), s_b = s(b,t), 0 \le t \le 1. \tag{3}$$

To facilitate the formulation of the desired approximation equation, Eq. 1 can be expressed in the form of

$$\frac{\partial^\alpha s}{\partial t^\alpha} = s^m \frac{\partial^2 s}{\partial x^2} + m s^{m-1}\left(\frac{\partial s}{\partial x}\right)^2. \tag{4}$$

To solve Eq. 4, the time-fractional derivative term of Eq. 4 can be approximated in the Caputo sense, that is [12],

$$\frac{\partial^\alpha s}{\partial t^\alpha} = \frac{1}{\Gamma(1-\beta)}\int_0^t \frac{\partial s}{\partial \varphi}(t-\varphi)^{-\alpha} d\varphi, \tag{5}$$

and the space derivative terms of Eq. 4 are discretized using first and second-order central difference schemes. The proposed method uses the usual finite difference framework in which the solutions, $s(x,t)$, $0 < x < x_f$ and $0 < t < t_f$, are partitioned uniformly with equidistant of both temporal and spatial steps, $h = x_f/M$, $M \in \mathbb{Z}^+$ and $k = t_f/N$, $N \in \mathbb{Z}^+$, respectively. The solutions can be computed using the discrete points $S_{p,n} = S(ph, nk)$ where for $p = 1, 2, \ldots, M-1$ and $n = 1, 2, \ldots, N$. Eq. 5 can be expressed in a discrete form as follows,

$$\frac{\partial^\alpha S_{p,n}}{\partial t^\alpha} = \tau \sum_{q=1}^{n} \sigma\left(S_{p,n-q+1} - S_{p,n-q}\right), \tag{6}$$

where

$$\tau = \tau(\alpha,k) = \frac{1}{\Gamma(1-\alpha)(1-\alpha)k^\alpha}, \tag{7}$$

and

$$\sigma = \sigma(\alpha,q) = q^{1-\alpha} - (q-1)^{1-\alpha}. \tag{8}$$

Using Eq. 6 together with the central difference schemes, an implicit finite difference approximation equation to represent Eq. 4 can be formulated into

$$\begin{aligned} \tau \sum_{q=1}^{n} \sigma\left(S_{p,n-q+1} - S_{p,n-q}\right) &= -\phi_1 S_{p,n}^m S_{p+1,n} + 2\phi_1 S_{p,n}^{m+1} - \phi_1 S_{p,n}^m S_{p-1,n} \\ &\quad - \phi_2 m S_{p,n}^{m-1} S_{p+1,n}^2 + 2\phi_2 m S_{p,n}^{m-1} S_{p+1,n} S_{p-1,n} - \phi_2 m S_{p,n}^{m-1} S_{p-1,n}^2, \end{aligned} \tag{9}$$

where $\phi_1 = 1/h^2$ and $\phi_2 = 1/(4h^2)$.

The approximate solutions of Eq. 4 can be obtained by solving the system of equations in the form of

$$\tilde{F} = \begin{bmatrix} F_{1,n} \\ F_{2,n} \\ \vdots \\ F_{M-1,n} \end{bmatrix}_{(M-1)\times 1} = 0, \tag{10}$$

where the dimension is $(M-1)\times 1$ for $p=1,2,\dots,M-1$ and $n=1,2,\dots,N$, and

$$F_{p,n}=c^{*}S_{p,n}-\phi_1 S_{p,n}^{m}S_{p+1,n}+2\phi_1 S_{p,n}^{m+1}-\phi_1 S_{p,n}^{m}S_{p-1,n}$$

$$-\phi_2 m S_{p,n}^{m-1}S_{p+1,n}^{2}+2\phi_2 m S_{p,n}^{m-1}S_{p+1,n}S_{p-1,n}-\phi_2 m S_{p,n}^{m-1}S_{p-1,n}^{2}-G_{p,n-1}. \tag{11}$$

Based on Eq. 11, c^{*} is a constant and $G_{p,n-1}$ is an accumulated value, which both are obtained from the calculation of

$$G_{p,n-1}=\tau\sum_{q=2}^{n}\sigma\left(S_{p,n-q+1}-S_{p,n-q}\right). \tag{12}$$

Next, when Newton's method is applied to solve Eq. 10, the resultant system of equations can be expressed in terms of

$$A\tilde{H}=-\tilde{F}, \tag{13}$$

and the paper suggests an iterative method to solve Eq. 13 due to its capability to solve Eq. 13 with better accuracy in which large matrices are taken per iteration. Also, the iterative method to be developed for solving Eq. 13 can find solutions in case the analytical method fails. Firstly, the development of the iterative method needs to consider the type of coefficient matrix A from the system of equations shown in Eq. 13. Since A is a tridiagonal matrix because of the result from the implicit finite difference discretization, it can be expressed as

$$A=\begin{bmatrix} D_1 & V_1 & & \\ L_2 & D_2 & V_2 & \\ & \ddots & \ddots & \ddots \\ & & L_{M-1} & D_{M-1}\end{bmatrix}_{(M-1)\times(M-1)}. \tag{14}$$

where the dimension $(M-1)\times(M-1)$ and the entries D_j, V_j and L_j for $j=1,2,\dots,M-1$ are calculated using

$$D_j=c^{*}-\phi_1 m S_j^{m-1}S_{j+1}+2\phi_1(m+1)S_j^{m}-\phi_1 m S_j^{m-1}S_{j-1}$$

$$-\phi_2 m(m-1)S_j^{m-2}S_{j+1}^{2}+2\phi_2 m(m-1)S_j^{m-2}S_{j+1}S_{j-1}$$

$$-\phi_2 m(m-1)S_j^{m-2}S_{j-1}^{2}, j=1,2,...,M-1, 0<c^{*}\le 1, \tag{15}$$

$$L_j=-\phi_1 S_j^{m}+2\phi_2 m S_j^{m-1}S_{j+1}-2\phi_2 m S_j^{m-1}S_{j-1}, j=2,...,M-1, \tag{16}$$

and

$$V_j=-\phi_1 S_j^{m}-2\phi_2 m S_j^{m-1}S_{j+1}+2\phi_2 m S_j^{m-1}S_{j-1}, j=1,2,...,M-2. \tag{17}$$

In addition, the values of $\tilde{H}$, which can be defined as the approximation of the solutions, are obtained by using

$$\tilde{H}=\tilde{S}^{(\ell)}-\tilde{S}^{(\ell-1)}, \ell=1,2,..., \tag{18}$$

which can also be expressed as

$$\tilde{H}=\begin{bmatrix} S_{1,n}\\ S_{2,n}\\ \vdots\\ S_{M-1,n}\end{bmatrix}^{(\ell)}_{(M-1)\times 1}-\begin{bmatrix} S_{1,n}\\ S_{2,n}\\ \vdots\\ S_{M-1,n}\end{bmatrix}^{(\ell-1)}_{(M-1)\times 1}. \tag{19}$$

Now, to derive the fractional Newton explicit group (FNEG) method for solving Eq. 10, let's say a system of four equations is taken randomly from Eq. 13, that is

$$L_iH_{i-1}+D_iH_i+V_iH_{i+1}=-F_i, \tag{20}$$

$$L_{i+1}H_i+D_{i+1}H_{i+1}+V_{i+1}H_{i+2}=-F_{i+1}, \tag{21}$$

$$L_{i+2}H_{i+1} + D_{i+2}H_{i+2} + V_{i+2}H_{i+3} = -F_{i+2}, \quad (22)$$

and

$$L_{i+3}H_{i+2} + D_{i+3}H_{i+3} + V_{i+3}H_{i+4} = -F_{i+3}, \quad (23)$$

which has the matrix form as

$$\begin{bmatrix} D_i & V_i & 0 & 0 \\ L_{i+1} & D_{i+1} & V_{i+1} & 0 \\ 0 & L_{i+2} & D_{i+2} & V_{i+2} \\ 0 & 0 & L_{i+3} & D_{i+3} \end{bmatrix} \begin{bmatrix} H_i \\ H_{i+1} \\ H_{i+2} \\ H_{i+3} \end{bmatrix} = - \begin{bmatrix} F_i + L_i H_{i-1} \\ F_{i+1} \\ F_{i+2} \\ F_{i+3} + V_{i+3}H_{i+4} \end{bmatrix}. \quad (24)$$

The inversion of Eq. 24 can yield the iterative formula of the FNEG method, that is, for $i = 1, 5, 9, \ldots,$

$$\begin{bmatrix} H_i \\ H_{i+1} \\ H_{i+2} \\ H_{i+3} \end{bmatrix}^{(\ell)} = -A^{*-1} \begin{bmatrix} F_i + L_i H_{i-1} \\ F_{i+1} \\ F_{i+2} \\ F_{i+3} + V_{i+3}H_{i+4} \end{bmatrix}^{(\ell-1)}. \quad (25)$$

where

$$A^* = \begin{bmatrix} D_i & V_i & 0 & 0 \\ L_{i+1} & D_{i+1} & V_{i+1} & 0 \\ 0 & L_{i+2} & D_{i+2} & V_{i+2} \\ 0 & 0 & L_{i+3} & D_{i+3} \end{bmatrix}. \quad (26)$$

This research study utilizes C++ programming language to code the FNEG method for solving TFPME. The main reason for using C++ is the flexibility of constructing iteration count and execution time for the FNEG implementation. The software used in this study is Dev-C++ by Embarcadero Technologies. The source code implementation uses a Lenovo laptop with the processor AMD Ryzen 7 5700U and 8 GB RAM. Below is the following algorithm of the FNEG method taken from the full source code.

Algorithm 1 FNEG method

while $n \leq N$ **do**
  Initialize $\ell_f = 0, \tilde{S}^{(\ell=0)} = 1.0$ and $\tilde{H}^{(\ell=0)} = 0$;
  Construct A^*;
  while $\left|\tilde{S}^{(\ell)} - \tilde{S}^{(\ell-1)}\right| > 10^{-10}$ **do**
    while $\left|\tilde{H}^{(\ell)} - \tilde{H}^{(\ell-1)}\right| > 10^{-10}$ **do**
      For $i = 1, 5, 9, \ldots$**;**

$$\begin{bmatrix} H_i \\ H_{i+1} \\ H_{i+2} \\ H_{i+3} \end{bmatrix}^{(\ell)} = -A^{*-1} \begin{bmatrix} F_i + L_i H_{i-1} \\ F_{i+1} \\ F_{i+2} \\ F_{i+3} + V_{i+3}H_{i+4} \end{bmatrix}^{(\ell-1)}.$$

      $\ell++$**;**
    end while
    $\tilde{S}^{(\ell)} = \tilde{H}^{(\ell)} + \tilde{S}^{(\ell-1)}$;
  end while
  $\ell_f = \ell_f + \ell$;
end while

3 Results and discussion

This study performed several repeated experiments on the proposed FNEG method by solving selected TFPME problems at various orders of time-fractional derivatives and matrix sizes. The first TFPME problem is given by

$$\frac{\partial^\alpha s}{\partial t^\alpha} = \frac{\partial}{\partial x}\left(s^m \frac{\partial s}{\partial x}\right), 0 < \alpha < 1, m \in \mathbb{N}, \quad (27)$$

subject to the initial condition

$$s_0 = s(x,0), 0 \leq x \leq 1, \tag{28}$$

Eq. 27 can be used to model the process of signal smoothing with an initial noisy signal given by Eq. 28 [1]. This problem considers a case of nonlinear signal diffusion with the function s represents the signal strength at a certain position and time. Here, the accuracy of the FNEG method is evaluated using an exact solution [13], that is

$$s(x,t) = x + \frac{t^{\alpha}}{\Gamma(1+\alpha)}. \tag{29}$$

Notice that when $\alpha = 1$, Eq. 29 becomes the solution of a porous medium equation [14],

$$s(x,t) = x + t. \tag{30}$$

The second problem is based on the equation that describes the fluid flow during the fingering phenomenon with an inclination and gravitational effect [15].

$$\frac{\partial^{\alpha} s}{\partial t^{\alpha}} = \frac{1}{4}\frac{\partial}{\partial x}\left(s^{3/2}\frac{\partial s}{\partial x}\right) - K\sin\theta s^2 \frac{\partial s}{\partial x}, 0 < \alpha \leq 0.5. \tag{31}$$

Here, the experiment uses random parameters $K = 0.25$ and $\theta = 0$. The experiment applies the same initial condition from Kesarwani and Meher [15], which is

$$s_0 = e^{-x}, 0 \leq x \leq 1, \tag{32}$$

and compare the numerical solutions with their analytical solution, which is given by

$$\begin{aligned} s = {} & e^{-x} + \left(-\frac{15}{8}h(e^{-x})^{5/2} - 2K\sin\theta(e^{-x})^3 - Kh\sin\theta(e^{-x})^3\right)\frac{ht^{\alpha}}{\Gamma(\alpha+1)} \\ & - \frac{3}{896}\frac{\left(\frac{25}{16}(e^{-x})^{5/2} + 3K\sin\theta(e^{-x})^3\right)^2\left(-2h(8K\sin\theta(e^{-x})^3 + 5(e^{-x})^{5/2})t^{\alpha}\right)^{7/2}}{\left(8K\sin\theta(e^{-x})^3 + 5(e^{-x})^{5/2}\right)^3\Gamma(\alpha+1)^{7/2}} \\ & - \frac{1}{3584}\frac{\left(-\frac{125}{32}(e^{-x})^{5/2} - 9K\sin\theta(e^{-x})^3\right)^3\left(-2h(8K\sin\theta(e^{-x})^3 + 5(e^{-x})^{5/2})t^{\alpha}\right)^{5/2}}{\left(8K\sin\theta(e^{-x})^3 + 5(e^{-x})^{5/2}\right)^2\Gamma(\alpha+1)^{5/2}} + \ldots, \end{aligned} \tag{33}$$

and $h = -0.035$ [15].

The numerical experiment of the FNEG method to solve the first problem is conducted using three different values of α such as 0.25, 0.50 and 0.75. Meanwhile, the experiment of solving the second problem considers $\alpha = 0.10, 0.30$ and 0.50. Both experiments use five different dimensions of matrix A, $(M-1)\times(M-1) = 256\times 256, 512\times 512, 1024\times 1024, 2048\times 2048$ and 4096×4096 to verify the numerical convergence of the solutions. The experiment evaluates the efficiency of the method based on the total iterations ℓ_f and C++ program time measured in seconds. Then, the accuracy of the method is judged according to the size of the absolute errors from comparing against available solutions, Eq. 29 and 33. A point Gauss-Seidel (GS) iterative method is applied to solve the selected problems as the benchmarking to the proposed FNEG method in terms of efficiency and accuracy. The experiment uses the point GS iterative method because it is commonly used to solve the system of equations arising from most fractional differential equations [16, 17, 18, 19]. The following Tables 1, 2, and 3 show the performance of the FNEG method against the benchmark method in solving the first problem at various α and A. On the other hand, Tables 4, 5 and 6 show the results of comparing the FNEG and benchmark methods from solving the second problem at different α and A.

Several significant findings were found based on the collected results from experimenting FNEG method to solve the two selected problems. Table 1 shows that the FNEG method needed lesser iterations than the GS method in computing numerical solutions of the first problem for all values of α and A. The C++ program elapsed time of FNEG was shorter than the GS method for all tests because the program time is highly correlated to the number of iterations, see Table 2. Then, as shown in Table 3, for the three values of α used in the first problem, the maximum absolute errors decrease when the sizes of matrices used in the computation increase. Therefore, the errors illustrated sufficient proof of numerical convergence of the finite difference scheme with Caputo's time fractional derivative operator.

The efficiency of the FNEG method was also proven to be better than the GS method when it was implemented to solve the second problem. Based on the results in Tables 4 and 5, the FNEG performed much lesser iterations and faster program time to obtain numerical solutions for the second problem. The absolute errors between the FNEG and GS methods are comparable for chosen values of α and A, see Table 6. Overall results indicated that the FNEG method via implicit finite difference scheme in the Caputo sense could be a good numerical method for solving TFPME for the one-dimensional case. Higher dimensional TFPME problems will be the subject of research interest in the near future.

Table 1: Total iterations to find solutions to the first problem at different α and A.

Method	α	A				
		256×256	512×512	1024×1024	2048×2048	4096×4096
GS	0.25	220,913	756,486	2,528,442	8,251,205	26,206,296
FNEG		65,624	226,941	776,679	2,597,848	8,432,295
GS	0.50	195,163	669,902	2,244,628	7,343,276	23,398,282
FNEG		57,640	199,635	685,268	2,298,473	7,484,378
GS	0.75	149,576	511,435	1,709,093	5,596,658	17,814,569
FNEG		43,976	152,853	523,157	1,746,948	5,679,010

Table 2: C++ program time (seconds) to complete all numerical solutions to the first problem at different α and A.

Method	α	A				
		256×256	512×512	1024×1024	2048×2048	4096×4096
GS	0.25	15.92	109.31	732.79	5374.00	35056.68
FNEG		8.39	55.41	371.44	2613.75	17382.81
GS	0.50	14.39	99.10	662.43	4942.83	32597.91
FNEG		7.56	50.09	335.63	2373.47	15947.89
GS	0.75	11.22	77.26	524.66	3917.38	25853.99
FNEG		5.85	38.55	258.52	1840.47	12474.91

Table 3: Maximum absolute errors from solving the first problem at different α and A.

Method	α	A				
		256×256	512×512	1024×1024	2048×2048	4096×4096
GS	0.25	2.7571E-4	2.7382E-4	2.6619E-4	2.3704E-4	1.2169E-4
FNEG		2.7618E-4	2.7567E-4	2.7376E-4	2.6619E-4	2.3564E-4
GS	0.50	6.5241E-4	6.5049E-4	6.4278E-4	6.1402E-4	5.0057E-4
FNEG		6.5288E-4	6.5236E-4	6.5044E-4	6.4276E-4	6.1200E-4
GS	0.75	1.3673E-3	1.3653E-3	1.3578E-3	1.3302E-3	1.2125E-3
FNEG		1.3678E-3	1.3673E-3	1.3653E-3	1.3575E-3	1.3274E-3

Table 4: Total iterations to find solutions to the second problem at different α and A.

Method	α	A				
		256×256	512×512	1024×1024	2048×2048	4096×4096
GS	0.10	192,608	656,839	2,187,946	7,104,066	20,171,159
FNEG		56,399	194,923	662,971	2,203,674	7,132,110
GS	0.30	174,209	596,419	1,998,331	6,542,402	20,787,358
FNEG		50,485	175,435	599,871	2,007,393	6,561,983
GS	0.50	154,768	532,165	1,795,258	5,934,675	19,128,587
FNEG		44,630	155,865	535,819	1,806,479	5,966,782

Table 5: C++ program time (seconds) to complete all numerical solutions to the second problem at different α and A.

Method	α	A				
		256×256	512×512	1024×1024	2048×2048	4096×4096
GS	0.10	2.48	11.25	64.87	834.04	32080.69
FNEG		2.20	8.16	41.70	251.81	1615.35
GS	0.30	2.36	10.53	60.87	387.85	2470.81
FNEG		2.16	7.73	39.41	234.97	1503.20
GS	0.50	2.25	9.75	55.39	359.41	2282.60
FNEG		2.09	7.38	35.57	212.15	1371.80

Table 6: Maximum absolute errors from solving the second problem at different α and A.

		A				
Method	α	256×256	512×512	1024×1024	2048×2048	4096×4096
GS	0.10	9.1623E-02	9.1626E-02	9.1630E-02	9.1641E-02	1.1399E-01
FNEG		9.1623E-02	9.1625E-02	9.1626E-02	9.1630E-02	9.1642E-02
GS	0.30	5.0628E-02	5.0630E-02	5.0632E-02	5.0637E-02	5.0654E-02
FNEG		5.0628E-02	5.0629E-02	5.0630E-02	5.0632E-02	5.0638E-02
GS	0.50	9.3209E-02	9.3203E-02	9.3198E-02	9.3189E-02	9.3160E-02
FNEG		9.3209E-02	9.3203E-02	9.3200E-02	9.3196E-02	9.3186E-02

4 Conclusion

This paper has presented an efficient iterative method called the fractional Newton explicit group for solving several time-fractional nonlinear porous medium equations. The finite difference approximation equation to the main problem is well derived based on implicit finite difference schemes with the Caputo time-fractional derivative operator. The novelty of this study is the systematic formulation of the proposed FNEG method using the implicit finite difference approximation in Caputo sense. The efficiency of the proposed FNEG method is evidently performs better than the Gauss-Seidel method for all kinds of problems, various α and A. In addition, the proposed FNEG method performs with higher accuracy for solving the first problem and comparable to the Gauss-Seidel method when solving the second problem. Since the proposed method is experimented via solving one-dimensional physics problems, the method's efficacy in solving realistic phenomena can be guaranteed. The next course of this study is to consider more complex problems, especially the higher dimensional problems. Future work can consider to improve the accuracy of the finite difference-based iterative method for solving higher dimensional time fractional nonlinear porous medium equations.

Acknowledgments We want to thank the Research Management Centre of University Malaysia Sabah, Malaysia, for their financial support in this work. This work is part of the research under the university research grant SBK0462-2021 the first author received.

References

[1] Y. Li, F. Liu, I. W. Turner and T. Li, Time-fractional diffusion equation for signal smoothing, *Appl. Math. Comput.*, **326**, 108–116 (2018).
[2] J. Kesarwani and R. Meher, Computational study of time-fractional porous medium equation arising in fluid flow through a water-wet porous media. *Int. J. Comput. Mater. Sci. Eng.***9** (2), (2020).
[3] F. Gharari, K. Arias-Calluari, F. Alonso-Marroquin and M. N. Najafi, Space-time fractional porous media equation: application on modeling of S& P500 price return. *Phys. Rev. E***104** (5), (2021).
[4] C. Imbert, R. Tarhini and F. Vigneron, Regularity of solutions of a fractional porous medium equation. *Inter. Free Bound.***22** (4), 401–442 (2020).
[5] B. Li, H. Wang and J. Wang, Well-posedness and numerical approximation of a fractional diffusion equation with a nonlinear variable order. *ESAIM: Math. Model. Numer. Anal.***55** (1), 171–207 (2021).
[6] N. A. Dao, Extinction in finite time of solutions to fractional parabolic porous medium equations with strong absorption. *Electron. J. Differ. Equ.***2021**, (2021).
[7] P. Wittbold, P. Wolejko and R. Zacher, Bounded weak solutions of time-fractional porous medium type and more general nonlinear and degenerate evolutionary integro-differential equations. *J. Math. Anal. Appl.***499** (1), (2021).
[8] Y. Yang and L. Wang, Lie symmetry analysis, conservation laws and separation variable type solutions of the time-fractional porous medium equation. *Waves Rand. Comp. Media* **32** (2), 980–999 (2020).
[9] Ł. Płociniczak, Numerical method for the time-fractional porous medium equation. *SIAM J. Numer. Anal.***57** (2), 638–656 (2019).
[10] J. Liu, X. Yang, Y. Feng and H. Zhang, Analysis of the time fractional nonlinear diffusion equation from diffusion process. *J. Appl. Anal. Comput.***10** (3), 1060–1072 (2020).
[11] H. Okrasińska-Płociniczak and Ł. Płociniczak, Second order scheme for self-similar solutions of a time-fractional porous medium equation on the half-line. *Appl. Math. Comput.***424**, (2022).
[12] I. Podlubny, *Fractional differential equations*, Academic Press, San Diego, 1999.
[13] H. A. Peker and F. A. Cuha, Application of Kashuri Fundo transform and homotopy perturbation methods to fractional heat transfer and porous media equations. *Therm. Sci.***26** (4A), 2877–2884 (2022).
[14] J. L. Vázquez, *The porous medium equation: mathematical theory*, Oxford University Press, New York, 2007.
[15] J. Kesarwani and R. Meher, Analytical study of time-fractional porous medium equation using homotopy analysis method. *AIP Conf. Proc.***2336**, (2021).
[16] I. Cilingir Sungu and H. Demir, A new approach and solution technique to solve time fractional nonlinear reaction-diffusion equations. *Math. Probl. Eng.***2015**, (2015).
[17] M. Sowmya and A. S. Vatsala, Generalized iterative methods for Caputo fractional differential equations via coupled lower and upper solutions with superlinear convergence. *Nonlin. Dyn. Syst. Theor.***15**(2), 198–208 (2015).
[18] D. Saikia, U. K. Saha and G. C. Hazarika, A numerical study of Atangana-Baleanu and Caputo-Fabrizio for MHD flow problem over a vertical hot stretching sheet with variable viscosity and thermal conductivity. *IAENG Int. J. Appl. Math.***50**(3), 616–627 (2020).
[19] A. Sunarto, P. Agarwal, J. Sulaiman and J. C. V. Lung, Numerical investigation on the solution of a space-fractional via preconditioned SOR iterative method. *Prog. Fract. Differ. Appl.***8** (2), 289–295 (2022).