Preconditioned Gauss-Seidel Method for the Solution of Time-fractional Diffusion Equations

A. Sunarto[1] and J. Sulaiman[2]
[1]Department Tadris Matematika, IAIN Bengkulu, Indonesia
[2]Department Matemathics with economics, Universiti Malaysia Sabah, Malaysia

Keywords: Linear System, Preconditioned Gauss-Seidel (PGS) method.

Abstract: In this paper, we deal with the application of an unconditionally implicit finite difference approximation equation of the one-dimensional linear time fractional diffusion equations via the Caputo's time fractional derivative. Based on this implicit approximation equation, the corresponding linear system can be generated in which its coefficient matrix is large scale and sparse. To speed up the convergence rate in solving the linear system iteratively, we construct the corresponding preconditioned linear system. Then we formulate and implement the Preconditioned Gauss-Seidel (PGS) iterative method for solving the generated linear system. One example of the problem is presented to illustrate the effectiveness of PGS method. The numerical results of this study show that the proposed iterative method is superior to the basic GS iterative method.

1 INTRODUCTION

Based on previous studies in (Mainardi, 1997; Diethelm and Freed, 1999; Meerschaert and Tadjeran, 2004; Zhang, 2009)many successful mathematical models, which are based on fractional partial derivative equations (FPDEs), have been developed. Following to that, there are several methods used to solve these models. For instance, we have transform method (Chaves, 1998), which is used to obtain analytical and/or numerical solutions of the fractional diffusion equations (FDE). Other than this method, other researchers have proposed finite difference methods such as explicit and implicit (Agrawal, 2002; Yuste and Acedo, 2005; Yuste, 2006). Also it is pointed out that the explicit methods are conditionally stable. Therefore, we discretize the time-fractional diffusion equation via the implicit finite difference discretization scheme and Caputo's fractional partial derivative of order α in order to derive a Caputo's implicit finite difference approximation equation. This approximation equation leads a tridiagonal linear system. Due to the properties of the coefficient matrix of the linear system which is sparse and large scale, iterative methods are the alternative option for efficient solutions.As far as iterative methods are concerned, it can be observed that many researchers such as Ghuang-hui (Ghuang-hui et al., 2006), Young (Young, 2014), Hackbusch (Hackbush, 1994) and Saad (Saad, 1996) have proposed and discussed several families of iterative methods. In addition to that, the concept of block iteration has also been introduced by Evans (Evans, 1985), Ibrahim and Abdullah (Ibrahim and Abdullah, 1995), Evans and Yousif (Evans, 1985) to demonstrate the efficiency of its computation cost. Among the existing iterative methods, the preconditioned iterative methods (Ghuang-hui et al., 2006), Zhao (Zhao et al., 2000), Hoang-hao (Hhonghao et al., 2009), Gunawardena (Gunawardena et al., 1991), Saad (Saad, 1996) have been widely accepted to be one of the efficient methods for solving linear systems.

Because of the advantages of these iterative methods, the aim of this paper is to construct and investigate the effectiveness of the Preconditioned Gauss-Seidel (PGS) iterative method for solving time fractional parabolic partial differential equations (TPPDE's) based on the Caputo's implicit finite difference approximation equation. To investigate the effectiveness of the PGS method, we also implement the Gauss Seidel (GS) iterative methods being used a control method.

To demonstrate the effectiveness of PGS method, let time fractional parabolic partial differential equation (TPPDE's) be defined as

Sunarto, A. and Sulaiman, J.
Preconditioned Gauss-Seidel Method for the Solution of Time-fractional Diffusion Equations.
DOI: 10.5220/0009882502680273
In *Proceedings of the 2nd International Conference on Applied Science, Engineering and Social Sciences (ICASESS 2019)*, pages 268-273
ISBN: 978-989-758-452-7
Copyright © 2020 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

$$\frac{\partial^\alpha U(x,t)}{\partial^\alpha} = \alpha(x)\frac{\partial^2 U(x,t)}{\partial x^2} + b(x)\frac{\partial U(x,t)}{\partial x} + c(x)U(x,t) \tag{1}$$

where $\alpha(x)$, $b(x)$ and $c(x)$ are known functions or constants, whereas α is a parameter which refers to the fractional order of time derivative.

The outline of this paper is organized as follows: In Section 2 and 3, an approximate the formula of the Caputo's fractional derivative operator and numerical procedure for solving time fractional diffusion equation (1) by means of the implicit finite difference method are given. In Section 4, formulation of the PGS iterative method is introduced. In Section 5 shows numerical example and its results and conclusion is given in Section 6.

2 PRELIMINARIES

Before constructing the Caputo's implicit finite difference approximation equation of Problem (1), the following are some basic definitions for fractional derivative theory which are used in this paper.

Definition 1. (Young, 2014) The Riemann-Liouville fractional integral operator, J^α of order$-\alpha$ is defined as

$$J^\alpha f(x) = \frac{1}{r(\alpha)}\int_v^x (x-t)^\alpha f(t)dt, \quad \alpha > 0x > 0 \tag{2}$$

Definition 2. (Young, 2014) The Caputo's fractional partial derivative operator, D^α of order$-\alpha$ is defined as

$$D^\alpha f(x) = \frac{1}{r(m-\alpha)}\int_0^x \frac{f^m(t)}{(x-t)^{\alpha-m+1}dt}, \alpha > 0 \tag{3}$$

with $m-1 < \alpha \leq m, m \in N, x > 0$

To obtain the numerical solution of Problem (1) with Dirichlet boundary conditions, firstly we derive an implicit finite difference approximation equation based on the Caputo's derivative definition and the non-local fractional derivative operator. This implicit approximation equation can be categoried as unconditionally stable scheme. To facilitate us in getting this approximation equation of Problem (1), let the solution domain of the problem be restricted to the finite space domain $0 \leq x \leq y$, with $0 < \alpha < 1$, whereas the parameter α refers to the fractional order of time derivative. In addition to that, consider boundary conditions of Problem (1) be given as

$$U(0,t) = g_0(t), U(l,t) = g_1(t),$$

and the initial condition

$$U(x,0) = f(x),$$

where $g_0(t), g_1(t)$, and f(x) are given functions. A discretize approximation to the time fractional derivative in Eq. (1) by using Caputo's fractional partial derivative of order α , is defined as (Young, 2014; Hackbush, 1994).

$$\frac{\partial u(x,t)}{\partial t^a} = \frac{1}{r(n-1)}\int_0^\infty \frac{\partial u(x-s)}{\partial t}(t-s)^{-\alpha} ds, t > 0, 0 < \alpha < 1 \tag{4}$$

3 APPROXIMATION FOR FRACTIONAL DIFFUSION EQUATION

According to Eq. (4), the formulation of Caputo's fractional partial derivative of the first order approximation method is given as

$$D_t^\alpha U_{i,n} \cong \sigma_{\alpha,k}\sum_{j=1}^{n}\omega_j^{(\alpha)} (U_{i,n} - j + 1 - U_{i,n-j}) \tag{5}$$

and we have the following expressions

$$\sigma_{\alpha,k} = \frac{1}{r(1-\alpha)(1-\alpha)k^\alpha}$$

and

$$\omega_j^{(\alpha)} = j^{1-\alpha}$$

Before discretizing Problem (1), let the solution domain of the problem be partitioned uniformly. To do this, we consider some positive integers m and n in which the grid sizes in space and time directions for the finite difference algorithm are defined as $h = \Delta x = \frac{\gamma-0}{m}$ and $k = \Delta t = \frac{T}{n}$ respectively. Based on these grid sizes, we construct the uniformly grid network of the solution domain where the grid points in the space interval $[0,\gamma]$ are indicated as the numbers $x_i = ih, i = 0,1,2,\ldots,m$ and the grid points in the time interval $[0,T]$ are labeled $t_j = jk, j = 0,1,2,\ldots,n$. Then the values of the function $U(x,t)$ at the grid point are denoted as $U_{i,j} = U(x_i,t_j)$.

SCIENCE AND TECHNOLOGY PUBLICATIONS

By using Eq. (5) and the implicit finite difference discretization scheme, the Caputo's implicit finite difference approximation equation of Problem (1) to the grid point centered at $(x_i, t_j) = (ih, nk)$ is given as

$$\sigma_{\alpha,k}\Sigma_{j=1}^{n}\omega_j^{(\alpha)}(U_{i,n-j+1} - U_(i,n-j)) = a_1\frac{1}{h^2}(U_{i-1,n} - 2U_{i,n} + U_{i+1,n}) + b_i\frac{1}{2h}(U_{i+1,n} - U_{i-1,n}) + c_iU_{i,n},) \quad (6)$$

for $i = 1, 2..., m-1$.

Based on Eq. (6), this approximation equation is known as the fully implicit finite difference approximation equation which is consistent first order accuracy in time and second order in space. Basically, the approximation equation (6) can be rewritten based on the specified time level. For instance, we have for $n \geq 2$:

$$\sigma_{/alpha,k}\Sigma_{j=1}^{n}\omega^{(/alpha)}(U_{i,n=j+1} - U_{i,n-j}) = (\tfrac{a_i}{h^2} - \tfrac{b_i}{2h})U_{i-1,n} + (c_i - \tfrac{2a_i}{h^2})U_{i,n} + (\tfrac{a_i}{h^2} - \tfrac{b_i}{2h})U_{i+1,n} \quad (7a)$$

$$\therefore \sigma_{/alpha,k}\Sigma_{j=1}^{n}\omega^{(/alpha)}(U_{i,n=j+1} - U_{i,n-j}) = p_iU_{i-1,n} + q_iU_{i,n} + r_iU_{i+1,n},$$

where

$$p_i = \frac{a_i}{h^2} - \frac{b_i}{2h}, q_i = c_i - \frac{2a_i}{h^2}, r_i = \frac{a_i}{h^2} - \frac{b_i}{2h}$$

Also, we get for $n = 1$,

$$-p_iU_{i-1,1} + q_i * U_{i,1} - r_iU_{i+1,1} = f_{i,1}, i = 1, 2, ..., m-1 \quad (7b)$$

where

$$\omega^{(/alpha)} = 1, q_i* = \sigma_{/alpha,k} - q_i, f_{i,1} = \sigma_{/alpha,k}U_{i,1}$$

Based on Eq. (7b), it can be seen that the tridiagonal linear system can be constructed in matrix form as

$$AU_{\backsim} = f_{\backsim} \quad (8)$$

where

$$A = \begin{bmatrix} q_1^* & -r_1 & & & & \\ -p_2 & q_2^* & -r_2 & & & \\ & -p_3 & q_3^* & -r_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -p_{m-2} & q_{m-2}^* & -r_{m-2} \\ & & & & p_{m-1} & -p_{m-1}^* \end{bmatrix}$$

(m-1)x(m-1)

$$U_{\backsim} = \begin{bmatrix} U11 & U_{21} & U_{31} & \cdots & U_{m-2,1} & U_{m-1,1} \end{bmatrix}^T$$

$$f_{\backsim} = \begin{bmatrix} U_{11} + p_1U_{01} & U_{21} & U_{31} & \cdots \\ & U_{m-2,1} & U_{m-1,1+p_{m-1}} & U_{m,1} \end{bmatrix}^T$$

4 FORMULATION OF PRECONDITIONED GAUSS-SEIDEL ITERATIVE METHOD

In relation to the tridiagonal linear system in Eq. (8), it is clear that the characteristics of its coefficient matrix are large scale and sparse. As mentioned in Section 1, many researchers have discussed various iterative methods such as Ghuang-Hui (Ghuang-hui et al., 2006), Zhao (Zhao et al., 2000), Hoang-hao (Hhong-hao et al., 2009), Gunawardena (Gunawardena et al., 1991), Young (Young, 2014), Hackbusch (Hackbush, 1994), Saad (Saad, 1996), Yousif and Evans (Evans and Yousif, 1986). To obtain numerical solutions of the tridiagonal linear system (8), we consider the Preconditioned Gauss-Seidel (PGS) iterative method (Ghuang-hui et al., 2006; Zhao et al., 2000; Hhong-hao et al., 2009; Gunawardena et al., 1991), which is the most known and widely using for solving any linear systems.

Before applying the PGS iterative method, we need to transform the original linear system (8) into the preconditioned linear system

$$A^*x_{\backsim} = f_{\backsim}^* \quad (9)$$

where

$$A^* = PAP^T,$$
$$f_{\backsim}^* = Pf_{\backsim}, U_{\backsim} = P^Tx_{\backsim}$$

Actually, the matrix P is called a preconditioned matrix and defined as (Kohno et al., 1997).

$$P = I + S$$

where

$$S = \begin{bmatrix} 0 & -r_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -r_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -r_3 & 0 & 0 \\ & & \ddots & \ddots & \ddots & \\ 0 & 0 & 0 & 0 & 0 & -r_{m-1} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{(m-1)x(m-1)}$$

and the matrix I is an identical matrix. To formulate PGS method, let the coefficient matrix in (8) be expressed as summation of the three matrices

$$A^* = D - L - V \tag{10}$$

where D, L and V are diagonal, lower triangular and upper triangular matrices respectively. By using Eq. (9) and (10) , the formulation of PGS iterative method can be defined generally as (Ghuang-hui et al., 2006; Zhao et al., 2000; Hhonghao et al., 2009; Gunawardena et al., 1991; Kohno et al., 1997).

$$\underset{\frown}{x}^{(k+1)} = (D-L)^{-1} V \underset{\frown}{x}^{(k)} + (D-L)^{-1} \underset{\frown}{f}^{*} \tag{11}$$

where $\underset{\frown}{x}^{(k+1)}$ represents an unknown vector at $(k+1)^{th}$ iteration. The implementation of the PGS iterative method can be described in Algorithm 1.

Algorithm 1: PGS method

i Initialize $U^{\frown} \leftarrow 0$ and $\varepsilon \leftarrow 10^{-10}$.

ii For $j = 1, 2, \cdots, n$Implement
For $i = 1, 2, \cdots, m-1$calculate

$$\underset{\frown}{x}^{(k+1)} = (D-L)^{-1} V \underset{\frown}{x}^{(k)} + (D-L)^{-1} \underset{\frown}{f}^{*}$$

$$\underset{\frown}{U}^{(k+1)} = P^T \underset{\frown}{x}^{(k+1)}$$

Convergence test. If the convergence criterion i.e $\left\| \underset{\frown}{U}^{(k+1)} - \underset{\frown}{U}^{(k)} \right\| \leq \varepsilon = 10^{-10}$ is satisfied, go to Step (iii). Otherwise go back to Step (a).

iii Display approximate solutions.

5 NUMERICAL EXAMPLE

By using approximation Eq.(7), we consider one example of the time fractional diffusion equation to test the effectiveness of the Gauss-Seidel (GS), and Preconditioned Gauss-Seidel (PGS) iterative methods. In order to compare the effectiveness of these two proposed iterative methods, three criteria have been considered such as number of iterations, execution time (in seconds) and maximum absolute error at three different values of $\alpha = 0.25, \alpha = 0.50$ and $\alpha = 0.75$. For implementation of both iterative schemes, the convergence test considered the tolerance error, which is fixed as $\varepsilon = 10^{-10}$.

Let us consider the time fractional initial boundary value problem be (Ali et al., 2013).

$$\frac{\partial^{/alpha} U(x,t)}{\partial t^{/alpha}} = \frac{\partial^2 U(x,t)}{\partial x^2}, 0 < /alpha \leq 1, 0 \leq x \leq y, t > 0 \tag{12}$$

where the boundary conditions are stated in fractional terms.

$$U_{(0,t)} = \frac{2kt^{/alpha}}{r(/alpha+1)}, U(l,t) = l^2 + \frac{2kt^{/alpha}}{r(/alpha+1)} \tag{13}$$

and the initial condition

$$U(x,0) = x^2 \tag{14}$$

From Problem (12), as taking $/alpha = 1$, it can be seen that Eq. (12) can be reduced to the standard diffusion equation

$$\frac{\partial U(x,t)}{\partial t} = \frac{\partial^2 U(x,t)}{\partial x^2}, 0 \leq x \leq \gamma, t > 0 \tag{15}$$

subjected to the initial condition

$$U(x,0) = x^2,$$

and boundary conditions

$$U(0,t) = 2kt, U(l,t) = l^2 + 2kt,$$

Then the analytical solution of Problem (15) is obtained as follows

$$U(x,t) = x^2 + 2kt.$$

Now by applying the series

$$U(x,t) = \Sigma_{n=0}^{m-1} \frac{\partial^n U(x,0)}{\partial t^n} \frac{t^n}{n!} + \Sigma_{n-1}^{\infty} \Sigma_{i=0}^{m-1} \frac{\partial^{mn+U} U(x,0)}{\partial t^{mn+i}} \frac{t^{n/alpha+i}}{r(n/alpha+i+1)}$$

to $U(x,t)$ for $0 < /alpha \leq 1$ it can be shown that the analytical solution of Problem (12) is given as

$$U(x,t) = x^2 + 2k \frac{t^{/alpha}}{r(/alpha+1)}.$$

All results of numerical experiments for Problem (12), obtained from implementation of GS and PGS iterative methods are recorded in Table 1 at different values of mesh sizes, m = 128, 256, 512, 1024, and 2048.

6 CONCLUSION

In order to get the numerical solution of the time fractional diffusion problems, the paper presents the derivation of the Caputo's implicit finite difference approximation equations in which this approximation equation leads a linear system. From observation of all experimental results by imposing the GS and PGS iterative methods, it is obvious at $/alpha = 0.25$ that number of iterations have declined approximately by 64.87-99.82% corresponds to the PGS iterative method compared with the GS method. Again in terms of execution time, implementations of PGS method are much faster about 4.96-93.03% than the GS method. It means that the PGS method requires the least amount for number of iterations and computational time at $/alpha = 0.25$ as compared with GS iterative methods. Based on the accuracy of both iterative methods, it can be concluded that their numerical solutions are in good agreement.

REFERENCES

Agrawal, O. (2002). Solution for a fractional diffusion-wave equation defined in a bounded domain. *Nonlinear Dynamics*, 29:145–155.

Ali, S., Ozgur, B., and Korkmaz, E. (2013). Analysis of fractional partial differential equations by taylor series expansion. *Boundary Value Problems*, page 68.

Chaves, A. (1998). Fractional diffusion equation to describe levy flight. *Physics Letters A*, 239:13–16.

Diethelm, K. and Freed, A. (1999). On the solution of nonlinear fractional order differential equation used in the modeling of viscoelasticity. In *Scientific Computing in Chemical engineering II Computational Fluid Dynamic, Reaction Engineering and Molecular Properties*, page 217–224. Spring, Heidelberg.

Evans, D. (1985). Group explicit iterative methods for solving large linear systems. *Int. J. Computer Maths*, 17:81–108.

Evans, D. and Yousif, W. (1986). Explicit group iterative methods for solving elliptic partial differential equations in 3-space dimensions. *Int. J. Computer Maths*, 18:323–340.

Ghuang-hui, C., Huang, T., and Cheng, X. (2006). The preconditioned gauss-seidel type iterative method for solving linear systems. *Applied Mathematics and Mechanics*, 27:1275–1279.

Gunawardena, A., Jain, S., and Synder, L. (1991). *Modified Iterative Methods for Consistent Linear Systems*. Elsevier Science Publishing Co.Inc.

Hackbush, W. (1994). *Iterative solution of large sparse systems of equations*. Springer, New York.

Hhonghao, H., Dongjin, Y., Yi, H., and Jinqiu, X. (2009). Preconditioned gauss-seidel iterative method for linear systems. *International Forum on Information Technology and Applications*, page 339.

Ibrahim, A. and Abdullah, A. (1995). Solving the two dimensional diffusion equation by the four point explicit decoupled group (edg) iterative method. *International Journal Computer Mathematics*, 58:253–256.

Kohno, T., Kotakemori, H., and Niki, H. (1997). Improving the modified gauss-seidel method for z-matrices. *Elsevier*, 267:113–123.

Mainardi, F. (1997). *Fractals and Fractional Calculus Continuum Mechanics*. Springer-Verlag, Heidelberg.

Meerschaert, M. and Tadjeran, C. (2004). Finite difference approximation for fractional advection-dispersion flow equations. *JCAM*, 172:145–155.

Saad, Y. (1996). *Iterative method for sparse linear systems*. International Thomas Publishing, Boston.

Young, D. (2014). *Iterative solution of large linear systems*. Elsevier, London.

Yuste, S. (2006). Weighted average finite difference method for fractional diffusion equations. *J.Comp.Phys*, 216:264–274.

Yuste, S. and Acedo, L. (2005). An explicit finite difference method and a new von neumann-type stability analysis for fractional diffusion equations. *SIAM Journal on Numerical Analysis*, 42:1862–1874.

Zhang, Y. (2009). A finite difference method for fractional partial differential equation, applied mathematics and computation. *Applied Mathematics and Computation*, 215:524–529.

Zhao, J., Zhang, G., Chang, Y., and Zhang, Y. (2000). A new preconditioned gauss-seidel method for linear systems, mathematics subject classification.

APPENDIX

Table 1: Comparison of number iterations, the execution time (seconds) and maximum errors for the iterative methods using example at $\alpha = 0.25, 0.50, 0.75$.

M	Method
128	**GS**
	PGS
256	**GS**
	PGS
512	**GS**
	PGS
1024	**GS**
	PGS
2048	**GS**
	PGS

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

Table 2: Comparison of number iterations, the execution time (seconds) and maximum errors for the iterative methods using example at $\alpha = 0.25, 0.50, 0.75$. (extension)

α = 0.25		
K	**Time**	**Max Error**
21017	37.73	9.97e-05
7292	35.86	9.96e-05
77231	343.63	1.00e-04
26884	261.56	9.98e-05
281598	2747.34	1.02e-04
98422	1916.28	1.00e-04
1017140	68285.36	1.09e-04
357258	14064.44	1.04e-04
3631638	158914.30	1.38e-04
21156	4104.17	1.36e-04

Table 3: Comparison of number iterations, the execution time (seconds) and maximum errors for the iterative methods using example at $\alpha = 0.25, 0.50, 0.75$ (extension).

α = 0.25		
K	**Time**	**Max Error**
13601	5.92	9.86e-05
4715	2.23	9.84e-05
50095	42.17	9.90e-05
17417	16.68	9.87e-05
183181	339.85	1.01e-04
63298	123.01	9.96e-05
663971	2454.53	1.08e-05
232784	1007.47	1.03e-05
2380946	17795.25	1.38e-04
19153.0	3239.84	134e-05

Table 4: Comparison of number iterations, the execution time (seconds) and maximum errors for the iterative methods using example at $\alpha = 0.25, 0.50, 0.75$ (extension).

α = 0.75		
K	**Time**	**Max Error**
6695	2.94	1.30e-04
2319	1.93	1.30e-04
24732	20.70	1.30e-04
8585	12.37	1.30e-04
90783	166.75	1.32e0-4
31619	62.78	1.31e-04
330622	1209.39	1.40e-04
115617	820.93	1.35e-04
1192528	8794.26	1.71e-04
12899	1305.5	1.35e-04