Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1, 896-911
2025
Publisher: Learning Gate
DOI: 10.55214/25768484.v9i1.4269

A Caputo-based nonlocal arithmetic-mean discretization for solving nonlinear time-fractional diffusion equation using half-sweep KSOR

Mohd Usran Alibubin[1], Jumat Sulaiman[2*], Fatihah Anas Muhiddin[3], Andang Sunarto[4], Graygorry Brayone Ekal[5]
[1,2,5]Faculty of Science and Natural Resources, Universiti Malaysia Sabah, Malaysia; mrusran@gmail.com (M.U.A.) jumat@ums.edu.my (J.S.)
[3]College of Computing, Informatics and Mathematics, Universiti Teknologi MARA Sabah Branch, Kota Kinabalu, Sabah, Malaysia; fatihahanas@uitm.edu.my (F.A.M.)
[4]Tadris Matematika, Universitas Islam Negeri (UIN) Fatmawati Sukarno, Indonesia; andang999@gmail.com (A.S.)

Abstract: This paper introduces a novel numerical method for solving one-dimensional nonlinear time-fractional diffusion equations (1DNTFDEs), addressing computational challenges in modeling nonlinearity and fractional dynamics. The proposed method integrates the Half-sweep Kaudd Successive Over-Relaxation (HSKSOR) technique with a Caputo-based nonlocal arithmetic-mean discretization scheme. The Caputo fractional derivative is leveraged to model time-fractional dynamics, while the half-sweep Caputo-based nonlocal arithmetic-mean scheme efficiently handles nonlinear terms, transforming the nonlinear system into a linear one solved iteratively using HSKSOR. Numerical experiments on three benchmark examples demonstrate significant reductions in iteration counts and computational time. The HSKSOR method outperforms traditional iterative techniques such as Full-Sweep Gauss-Seidel (FSGS) and Full-Sweep Kaudd Successive Over-Relaxation (FSKSOR) methods, achieving superior computational efficiency without sacrificing accuracy. The proposed method provides an efficient and scalable computational framework for solving complex time-fractional models, offering high accuracy and substantial computational cost reductions. This advancement enhances the theoretical framework of nonlocal discretization and offers a powerful tool for applications in physics, engineering, and applied mathematics, where modeling fractional dynamics is critical.

Keywords: *Caputo Nonlocal Finite Difference Scheme, Fractional Calculus, Iterative Solvers, Half-sweep KSOR Method, One-Dimensional Nonlinear Time-Fractional Diffusion Equations.*

1. Introduction

One-dimensional nonlinear time-fractional diffusion equations (1DNTDEs) have become an essential tool for modeling complex physical phenomena, particularly in anomalous diffusion [1]. Unlike classical diffusion, governed by integer-order derivatives, anomalous diffusion captures nonlocal dynamics and memory effects through fractional derivatives. These models are applied in diverse fields such as fractional kinetics [1] viscoelastic material analysis [2] fluid mechanics [3, 4] and image processing [5] where processes deviate from standard diffusion [1, 2]. Fractional differential equations (FDEs) are crucial for describing these anomalous transport phenomena, significantly extending traditional integer-order models [6, 7]. Key contributions from researchers such as Wang and Ren [8]; Wu, et al. [9]; Sunarto, et al. [10] and Sunarto, et al. [11] have advanced our understanding of fractional derivatives, particularly in their ability to model memory effects and long-range interactions in complex systems. This progress has sparked growing interest in time-fractional diffusion equations (TFDEs), which effectively simulate the dynamics of such systems.

Recent researchers have increasingly focused on the computational approach for solving both linear and nonlinear FDEs, especially TFDEs, due to their ability to model non-classical behaviors

History: Received: 5 December 2024; Revised: 6 January 2025; Accepted: 9 January 2025; Published: 15 January 2025
* Correspondence: jumat@ums.edu.my

in real-world systems Podlubny [12]. Fractional operators, such as the Caputo derivative, are instrumental in understanding complex mechanical and physical processes, including non-Markovian random walks and long-term memory effects [13]. However, analytical solutions to FDEs are often difficult to obtain, with only a few solvable using special functions such as the Mittag-Leffler function [14] the H-function [15] and the Wright function [16]. As a result, a range of numerical methods has been developed to overcome the challenges associated with solving fractional-order systems. Among the approaches gaining significant attention in the literature are the reduced spline (RS) method based on proper orthogonal decomposition (POD) [17, 18] the Crank-Nicholson finite element strategy [19, 20] and innovative approaches like the fractional finite differences [21] localized radial basis functions (RBFs) [22] and fractional differential quadrature (FDQ) [23].

The main numerical challenge of TFPDEs is solving a large and sparse system of linear equations (SLEs). Traditional methods, such as the Gauss-Seidel (GS) iterative technique, often exhibit slow convergence, making them less efficient for complex systems. Recently, the Kaudd Successive Over-Relaxation (KSOR) method has shown promise as a more efficient alternative, particularly for accelerating convergence in large linear systems derived from the discretization of FDEs [24, 25]. Despite significant research on iterative techniques for integer-order equations, their application to FDEs remains underexplored [26, 27]. While many studies have employed implicit schemes [28, 29] and the Caputo fractional derivative for solving TFPDEs, most have focused on linear problems, leaving a gap in the exploration of efficient iterative methods for NTFDEs. Additionally, most studies address the discretization of fractional derivatives using finite difference (FD) schemes, which generate large linear systems requiring effective solution techniques. The KSOR method, a variant of the SOR method, offers potential improvements in convergence speed due to its adaptive over-relaxation parameter. However, limited research has investigated its application to NTFDEs, particularly in comparison to traditional methods like GS [30, 31].

This paper aims to evaluate the performance of the HSKSOR method, which employs half-sweep iteration, for solving 1DNTFDEs using the Caputo fractional derivative. A half-sweep Caputo-based nonlocal arithmetic-mean discretization (NAMD) scheme is utilized to transform the nonlinear system into an equivalent linear system, which is then solved using the HSKSOR method. This study benchmarks HSKSOR against the Full-Sweep GS (FSGS) and Full-Sweep KSOR (FSKSOR) methods in terms of iteration counts, computational time, and accuracy, with a specific focus on the nonlocal and nonlinear properties of the NTFDEs. By conducting this comparison, the paper demonstrates the advantages of HSKSOR over FSKSOR and FSGS in solving large and sparse systems from the discretization of 1DNTFDEs, contributing to the development of more advanced numerical methods for FDEs.

The paper is organized as follows: Section 2 presents the formulation of the 1DNTFDEs, including the governing equations, boundary conditions, and the discretization process using the Caputo fractional derivative. Section 3 presents the numerical methods, mainly the application of the HSKSOR and FSGS techniques to solve the resulting linear systems arising from the discretization of the NTFDEs. Section 4 provides the results and discussion, comparing the performance of the HSKSOR, FSKSOR and FSGS methods in terms of iteration counts, computational time, and accuracy. Finally, Section 5 concludes with a summary of the key findings and offers directions for future research.

2. Fundamental of Fractional Calculus Concepts

To initiate the derivation of the HSKSOR iterative method, it is essential to introduce the fundamental concepts of fractional differential equations. A general form of 1DNTFDEs can be expressed as:

$$\frac{\partial^{\alpha} u(x,t)}{\partial t^{\alpha}} = \gamma \frac{\partial^{2} u(x,t)}{\partial x 2} - \beta u(1-u), \quad x \in [\rho_0, \rho_1], \quad 0 \le t \le T \tag{1}$$

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

In Eq. (1), $u(x,t)$ represents the unknown function, γ denotes the diffusion coefficient, and $u(1-u)$ defines the nonlinear term, with the fractional-order time derivative α constrained to $0 < \alpha \leq 1$. The spatial domain is defined by $x \in [\rho_0, \rho_1]$ and $0 \leq t \leq T$. Before introducing the proposed Caputo-based NAMD for Problem (1), it is essential to review the various approaches available for defining fractional calculus [26, 27].

Definition 1. The Fractional Integral operator Rieman-Liouville with order positive α is formulated as follows [28]:

$$J^{\alpha} f(x) = \frac{1}{\Gamma(\alpha)} \int_0^x (x-t)^{\alpha} f(t) dt, \quad \alpha > 0, \quad x > 0 \tag{2}$$

Definition 2. The Caputo fractional partial derivative with order α is formulated as follows [28]:

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

With $m-1 < \alpha \leq m, m \in \mathbb{N}, x > 0$. In Equations (2) and (3), where $\Gamma(\alpha)$ is the gamma function

$$\Gamma(\alpha) = \int_0^{\infty} x^{t-1} e^{-x} dx. \tag{4}$$

The Caputo derivative is advantageous because it reduces to the classical first derivative when $\alpha = 1$, and it facilitates the use of standard boundary and initial conditions. The memory effect captured by the Caputo derivative makes it suitable for modeling TFDEs, where the current state of the system depends on its entire history.

To solve the NTFDEs in Eq. (1), we employ numerical methods utilizing the Caputo derivative, incorporating Dirichlet boundary conditions, and considering the nonlocal fractional derivative operator. This approximation is categorized as unconditionally stable. For Problem (1), the solution is confined to a finite spatial domain $0 \leq x \leq \gamma$, with $0 < \alpha < 1$, where α represents the order of the spatial fractional derivative.

Let us consider the following initial and boundary conditions for Problem (1):

$$u(\rho_0, t) = g_0(t), \quad u(\rho_1, t) = g_1(t), \quad 0 \leq t \leq T \tag{5}$$

with initial condition

$$u(x,0) = f(x), \quad x \in [\rho_0, \rho_1] \tag{6}$$

where $g_0(t)$ and $g_1(t)$ are predefined functions that represent the fixed boundary values of u. Using the discretized approximation of the NTFDEs in Eq. (1), we apply Caputo fractional partial derivative of order α as follows:

$$\frac{\partial^{\alpha} u(x_i, t_n)}{\partial x^{\alpha}} = \frac{1}{\Gamma(n-\alpha)} \int_0^{\infty} \frac{\partial u(x-s)}{\partial t} (t_n - s)^{-\alpha} ds, \quad t > 0, 0 < \alpha < 1. \tag{7}$$

3. Caputo-Based Nonlocal Arithmetic-Mean Approximation

In this section, we present the proposed approach for solving NTFDE using the Caputo-based NAMD scheme combined with the HSKSOR technique. The method offeres an efficient numerical framework for solving large-scale nonlinear systems by reducing computational complexity while maintaining accuracy.

Based on Eq. (7), the formulation of Caputo fractional partial derivative for the first-order approximation is given by Murio [34]:

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

$$D_t^{\alpha} U_{i,n} \cong \sigma_{\alpha,k} \sum_{j=1}^{n} \omega_j^{(\alpha)} \left(U_{i,n-j+1} - U_{i,n-j} \right) \tag{8}$$

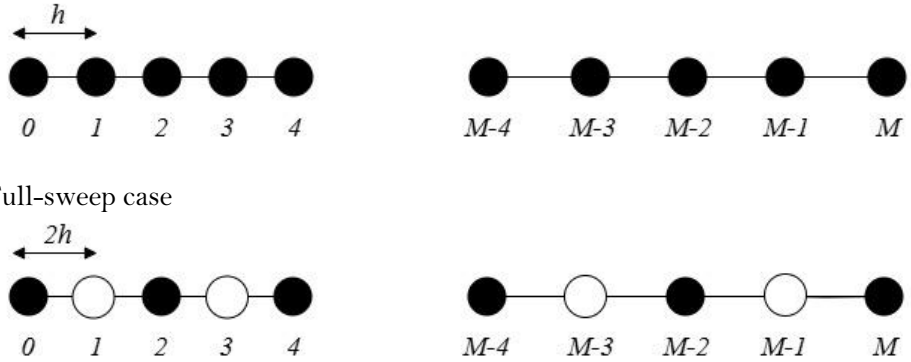
We also have the following expressions:

$$\sigma_{\alpha,k} = \frac{1}{\Gamma(1-\alpha)(1-\alpha)^{1-\alpha}}, \tag{9}$$

and

$$\omega_j^{(\alpha)} = j^{1-\alpha} - (j-1)1^{1-\alpha}. \tag{10}$$

The first step in discretizing Problem (1) is to partition the solution domain uniformly. Let m and n represent positive integers such that the grid sizes in space and time for the finite-difference scheme are given by $h = \delta x = \gamma/m$ and $k = \delta t = T/n$, respectively. Using these values, we construct a uniform grid across the solution domain. Spatial grid points in the interval $[0, \gamma]$ are defined as $x_i = ih, i = 0, 1, 2, \ldots, m$, while the time grid points in $[0, T]$ are expressed as $t_j = jk, j = 0, 1, 2, \ldots, n$. The function $U(x, t)$ at each grid point is written as $U_{i,j} = U(x_i, t_j)$, serving as the basis for the numerical approximation. The distribution of the interior node points across the finite grid is depicted in Figure 1.



(a) Full-sweep case

(b) Half-sweep case

Figure 1.
Distribution of the uniform mesh size for full- and half-sweep cases

By applying Eq. (8) along with the implicit finite difference discretization scheme, the Caputo implicit finite difference approximation at the grid point $x_i, t_j = (ih, jk)$ for Problem (1) is expressed as:

$$\begin{aligned} &\sigma_{\alpha,k} \sum_{j=1}^{n} \omega_j^{(\alpha)} \left(U_{i,n-j+1} - U_{i,n-j} \right) \\ &= \gamma \left(\frac{U_{i-2,n} - 2U_{i,n} + U_{i+2,n}}{4h^2} \right) + \beta U_{i,n} \left(1 - U_{i,n} \right) \end{aligned} \tag{11}$$

For $i = 2,4 \ldots, m-2$, the approximation achieves first-order accuracy in time and second-order accuracy in space, as shown in Eq. (11). To further develop the Caputo-based NAMD for Eq. (1), several half-sweep Caputo NAMD methods are considered, as discussed in Li, et al. [31] and Wang and Wang [32].

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

$$U_{i,n}^{2} \cong U_i + U_{i+2} \tag{12}$$

$$U_{i,n}^{2} \cong \left(\frac{U_i - 2,n + U_{i+2,n}}{2}\right) U_{i,n} \tag{13}$$

$$U_{i,n}^{3} \cong \left(\frac{U_i - 2,n + U_{i+2,n}}{2}\right) U^{2}{}_{i,n} \tag{14}$$

The Caputo-based nonlocal arithmetic-mean scheme from Eq. (13) is employed to transform the corresponding nonlinear Caputo-based nonlocal arithmetic-mean approximation equation into a system of linear equations (SLEs). To achieve this, we define intermediate variables and apply the Caputo-based nonlocal arithmetic-mean linearization derived from Eq. (13), leading to:

$$\beta_i = 1 - u_{i,n} = 1 - \left(\frac{u_{i-2,n} + u_{i+2,n}}{2}\right) \tag{15}$$

This result is then applied to Eq. (11):

$$\sigma_{\alpha,k} \sum_{j=1}^{n} \omega_j^{(\alpha)} \left(U_{i,n-j+1} - U_{i,n-j}\right) = a_0 \left(U_{i-2,n} - 2U_{i,n} + U_{i+2,n}\right) + \beta U_{i,n}\left(\beta_i\right) \tag{16}$$

The approximation equation can be adapted based on the chosen time level. For example, when $n \geq 2$, the structure of the equation adjusts to account for higher time levels, as shown below:

$$\sigma_{\alpha,k} \sum_{j=1}^{n} \omega_j^{(\alpha)} \left(U_{i,n-j+1} - U_{i,n-j}\right) = a_0 U_{i-2,n} + \left(p_i\right) U_{i,n} + a_0 U_{i+2,n} \tag{17}$$

By simplifying Eq. (17) for $n = 1$, the approximation equation can be expressed as:

$$\sigma_{\alpha,k}\left[U_{i,1} - U_{i,0}\right] = a_0 U_{i-2,1} - p_i U_1 + a_0 U_{i+2,1} \tag{18}$$

The approximate Eq. (18) can be rewritten as:

$$-U_{i-2,1} - q_i U_{i,1} - U_{i+2,1} = f_i U_{i,0} \tag{19}$$

where

$$q_i = \frac{p_i}{a_0},\ f_i = \frac{\sigma_{\alpha,k}}{a_0},\ a_0 = \frac{\gamma}{4h^2},\ p_i = \sigma_{\alpha,k} + 2a_0 + \beta\beta_i,\ \beta_i = 1 - \left(\frac{U_{i-2,n} + U_{i+2,n}}{2}\right)$$

Finally, by reordering the terms, the approximation equation is rewrittem in matrix form as follows:

$$A\tilde{U} = \tilde{f} \tag{20}$$

where

$$A = \begin{bmatrix} q_2 & -1 & & & & \\ -1 & q_4 & -1 & & & \\ & -1 & q_6 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & q_{m-4} & -1 \\ & & & & -1 & q_{m-2} \end{bmatrix}_{\left(\left(\frac{m}{2}\right)-1\right)\times\left(\left(\frac{m}{2}\right)-1\right)},$$

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

$$\underset{\sim}{U} = \begin{bmatrix} U_{2,1} & U_{4,1} & U_{6,1} & \cdots & U_{m-4,1} & U_{m-2,1} \end{bmatrix}^T,$$

$$\underset{\sim}{f} = \begin{bmatrix} U_{2,1}+p_1U_{01} & U_{4,1} & U_{6,1} & \cdots & U_{m-4,1} & U_{m-2,1}+p_{m-2}U_{m,1} \end{bmatrix}$$

4. Formulation of Half-Sweep Kaudd Successive Over-Relaxation (HSKSOR)

The linear system described by Eq. (20) involves a coefficient matrix that is large-scale and sparse, which presents a well-known challenge in numerical differential equations solutions. As outlined in Section 1, numerous researchers, such as Alibubin, et al. [33]; Murio [34]; Young [18] and Sabdin, et al. [35] have explored various iterative methods to address such systems. In this study, we employ the HSKSOR iterative method, which is well-regarded for its effectiveness and reduced the computational complexity in solving linear systems.

To implement the HSKSOR method, it is essential to reformulate the original linear system in Eq. (20) into a suitable diagonal form, thereby enabling the efficient application of the iterative scheme. This transformation is critical to fully harnessing the computational advantages of the HSKSOR method, particularly in terms of enhanced convergence rate and performance when dealing with large-scale and sparse matrices. The HSKSOR method is derived by decomposing the coefficient matrix A in Eq. (20) as follows:

$$A = D + L + V \tag{21}$$

In this equation, D, L, and V represent the diagonal, lower triangular, and upper triangular matrices, respectively. This decomposition allows for the development of general iterative methods. Both the FSKSOR and the HSKSOR methods can be formulated in matrix form based on this decomposition, as explained in previous studies [33, 36].

$$\underset{\sim}{U}_j^{(k+1)} = \left(\left(1-\omega^*\right)D-\omega^*L\right)^{-1}\left(D+\omega^*V\right)+\left(\left(1-\omega^*\right)D-\omega^*L\right)^{-1}\left(\omega^*\underset{\sim}{f}_j\right) \tag{22}$$

In Eq. (22), $u_j^{(k+1)}$ represents the unknown vector at $(k+1)-th$ iteration, where j refers to the grid index. The relaxation parameter ω, with its typical range $\omega \in (-2,0)$, plays a key role in determining the convergence behavior of the iterative method. The value of ω influences the rate at which the solution coverges to the result, with the optimal choice ensuring faster convergence. Algorithm 1 provides the details for implementing the HSKSOR method.

Algorithm 1: HSKSOR

- Initialize $\underset{\sim}{U}_j^{(k)} = 0$ and convergence tolerance $\varepsilon = 10^{-10}$
- Determine the optimal relaxation parameter ω
- For $i = 2,4 \ldots, m-2$, Use Eq. (23) to update unknowns at each grid point.
- Perform the convergence test. If the convergence criterion i.e., $\left\| \underset{\sim}{U}^{(k+1)} - \underset{\sim}{U}^{(k)} \right\| \leq \varepsilon = 10^{-10}$is satisfied, proceed to step (v). Otherwise, repeat the step (iii).
- Do direct calculation for the remaining node points.
- Showcase the numerical results.

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

5. Numerical Experiment

This section presents numerical experiments designed to evaluate the performance of the HSKSOR method in solving a linear system derived from the Caputo-based NAMD. The experiments, performed across varying mesh sizes, assessed the methods's computational efficiency. Key performance metrics, including iteration count (K), computational time (t), maximum error, and L_2-norm error, were used to benhcmark the proposed method against FSKSOR and FSGS. The results demonstrate the superior efficiency of the HSKSOR method in handling the nonlocal and nonlinear characteristics of NTFDEs. All computations were implemented in C, with error norms computed following methodologies from Alibubin, et al. [37] and Rahman, et al. [30].

Example 1:

We consider the following NTFDE with Caputo time-fractioanl derivatives and nonlinear terms [29]:

$$\frac{\partial^{\alpha}u(x,t)}{\partial t^{\alpha}}=\gamma\frac{\partial^{2}u(x,t)}{\partial x^{2}}+Au(1-u),\quad 0<\alpha\le 1,0\le x\le 1,0<t\le T \tag{23}$$

where $A=1$ and the boundary conditions are stated in fractional terms

$$u(0,t)=\frac{1}{\left(1+e^{-\frac{5}{6}t}\right)^{2}},\quad u(1,t)=\frac{1}{\left(1+e^{\frac{1}{\sqrt{6}}-\frac{5}{6}t}\right)^{2}}, \tag{24}$$

and the initial condition for $u(x,0)$ is derived from the exact solution:

$$u(x,0)=\frac{1}{\left(1+e^{\sqrt{\frac{x}{6}}}\right)^{2}} \tag{25}$$

The exact solution for the Eq. (23) is given by:

$$u(x,t)=\frac{1}{\left(1+e^{\frac{x}{\sqrt{6}}-\frac{5}{6}t}\right)^{2}} \tag{26}$$

Example 2: [38]

We consider the following NTFDE with Caputo time-fractional derivatives and nonlinear terms:

$$\frac{\partial^{\alpha}u(x,t)}{\partial t^{\alpha}}=\gamma\frac{\partial^{2}u(x,t)}{\partial x^{2}}+6u(1-u),\quad 0<\alpha\le 1,0\le x\le 1,0<t\le T \tag{27}$$

where γ is a diffusion coefficient and $\lambda=1$ represents the reaction term coefficient and the boundary conditions are stated in fractional terms

$$u(0,t)=\frac{1}{\left(1+e^{-5t}\right)^{2}},\quad u(1,t)=\frac{1}{\left(1+e^{1-5t}\right)^{2}}, \tag{28}$$

and the initial condition for $u(x,0)$ is derived from the exact solution:

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

$$u(x,0)=\frac{1}{\left(1+e^{x}\right)^{2}} \tag{29}$$

The exact solution for the Eq. (27) is given by:

$$u(x,t)=\frac{1}{\left(1+e^{x-5t}\right)^{2}} \tag{30}$$

Example 3: [30]

NTFDE characterized by the following equation:

$$\frac{\partial^{\alpha}u(x,t)}{\partial t^{\alpha}}=\gamma\frac{\partial^{2}u(x,t)}{\partial x^{2}}+\lambda u(1-u),\quad 0<\alpha\le 1,0\le x\le 1,0<t\le T \tag{31}$$

where γ is a diffusion coefficient, $\lambda=1$ represents the reaction term, and $\beta={}^{2}/_{3}$ is the initial value constant. The boundary conditions can be inferred from the exact solution of Eq. (31). By evaluating the boundaries$x=0$ and $x=1$, we find that the solution remains independent of x. Therefore, no explicit boundary conditions are required for this problem.

The initial condition is defined as:

$$u(x,0)=\beta=\frac{2}{3} \tag{32}$$

The exact solution for Eq. (32) is given by:

$$u(x,t)=\frac{\beta e^{t}}{1-\beta+\beta e^{t}},\quad \beta=\frac{2}{3} \tag{33}$$

Tables 1 to 9 present the complete numerical experiment for equations (23), (27) and (31), achieved through the application of the FSGS, FSKSOR and HSKSOR iterative methods. The experiments were conducted across a range of mesh sizes (m = 512, 1024, 2048, 4096, and 8192), providing a detailed evaluation of the comparative effectiveness of these techniques.

6. Discussion

The numerical results presented in Tables 1 to 3 offer a comprehensive comparison of the FSGS, FSKSOR, and HSKSOR methods for solving NTFDEs with fractional orders α=0.333,0.666, and 0.999. These results, evaluated at various mesh sizes (M=512, 1024,2048,4096, and 8192), reveal significant differences in terms of computational performance and accuracy across the methods. A key finding is the superior computational efficiency of the HSKSOR method, particularly for large-scale problems. The iteration counts (K) and computational times (t) for FSGS grow exponentially with increasing mesh size. For example, at M=8192 and α=0.333 (Table 1), FSGS required 24,114,142 iterations, taking 82,862.28 seconds, whereas HSKSOR required only 10,871 iterations and 127.76 seconds. Similarly, in Table 2 (α=0.333), HSKSOR completed the computation in 133.59 seconds with 11,250 iterations, while FSGS needed 24,637,082 iterations and over 84,000 seconds.

These results underscore that HSKSOR significantly reduces computational cost compared to both FSGS and FSKSOR. The method's key advantage lies in updating only half of the grid points per iteration, effectively reducing the iteration count and computational time without compromising accuracy, also see Figures. 2 to 4. These figures confirm that despite the computational efficiency gains provided by HSKSOR, the accuracy metrics remain consistent across the various fractional orders and mesh sizes, further demonstrating the robustness of the method. This makes HSKSOR particularly well-suited for large-scale, high-resolution fractional diffusion problems.

The effect of the fractional order α on computational performance is also evident. Lower values of α (e.g., α=0.333) correspond to stronger memory effects and thus require more iterations and

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

computational time. As α approaches 1, the problem becomes closer to a classical diffusion equation, reducing the iteration count. For instance, at M=8192, FSGS required 24,114,142 iterations for α=0.333 but only 741,257 iterations α=0.999 (see in Table 1). Similarly, HSKSOR required 10,871 iterations for M=8192 when α=0.333 but only 2,690 iterations for α=0.999. This demonstrates HSKSOR's efficiency across a range of fractional orders, highlighting its robustness in solving NTFDEs, regardless of memory effects or diffusion behavior. Additionally, we computed the percentage decrease obtained with the FSKSOR and HSKSOR methods compared to those from the FSGS method, reaching up to 99%, as shown in Table 4-6. These results clearly show that the HSKSOR method, based on the Caputo-based NAMD scheme, is the most effective approach for problem (1) out of the three methods.

7. Conclusion

This study highlights the successful application of the HSKSOR iterative method to efficiently solve a linear system derived from the Caputo-based nonlocal arithmetic-mean approximation equation. Numerical experiments, summarized in Tables 1 to 3, clearly demonstrate the method's computational advantages over both FSGS and FSKSOR, while ensuring accuracy. HSKSOR significantly reduces iteration counts by 99.8% to 99.9% compared to FSGS, depending on the mesh size and fractional order. For example, at M=8192 and α=0.333, iteration count reductions exceeded 99.95%. When compared to FSKSOR, iteration count reductions ranged between 45% to 51%, reflecting HSKSOR's ability to nearly halve the computational steps with no loss in accuracy. Similarly, in terms of computational time, HSKSOR proved highly efficient. Time reductions of 99.8% and 99.9% were achieved relative to FSGS. For instance, computational time dropped from 82,862.28 seconds (FSGS) to 127.76 seconds at M=8192 and α=0.333, a reduction of 99.85%. When compared to FSKSOR, time reductions ranged from 67% to 75%, further comfirming HSKSOR's efficiency. Importantly, these significant gains in efficiency did not come at the cost of accuracy.

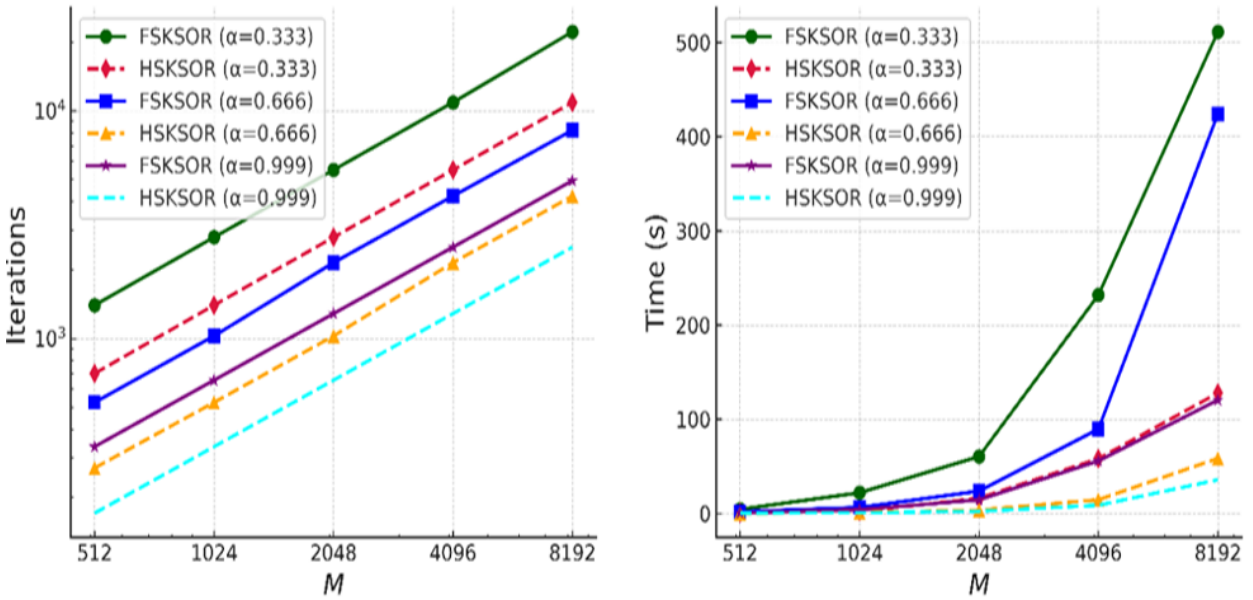


Figure 2.
Iteration count and computation time (seconds) plotted against M of two different approach based on three alpha values for Example 1.

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

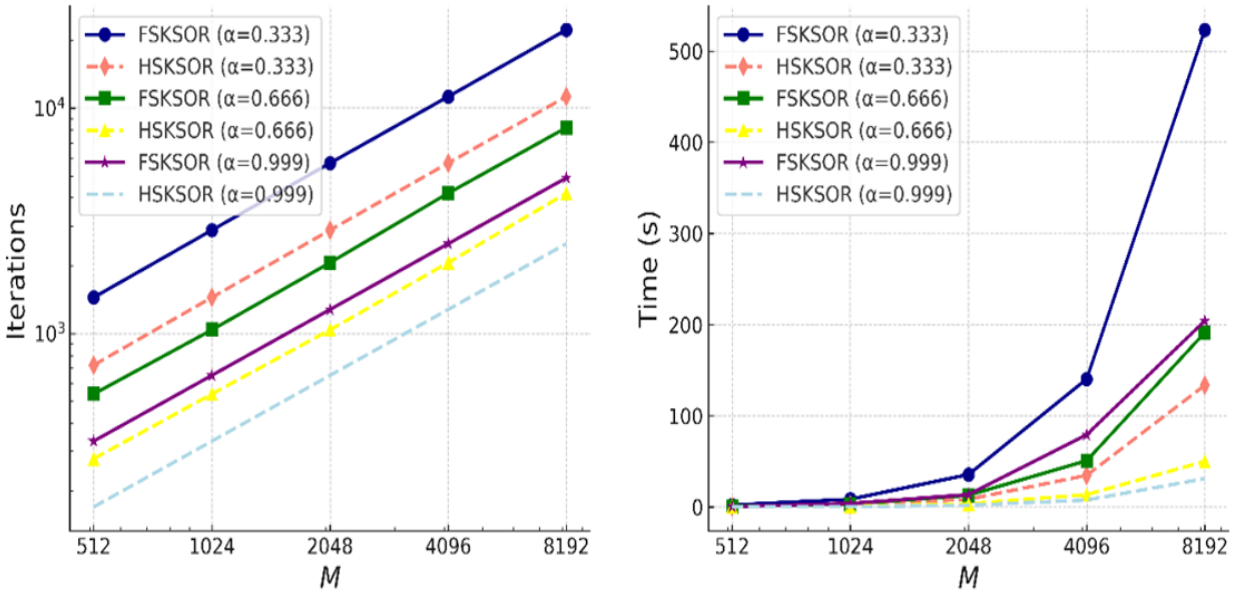


Figure 3.
Iteration count and computation time (seconds) plotted against M of two different approach based on three alpha values for Example 2.

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

Table 1.
The computed numerical result for Example 1 at $\alpha = 0.333, 0.666$ and 0.999.

M	Method	$\alpha = 0.333$				$\alpha = 0.666$				$\alpha = 0.999$			
		K	t	Max. E	L2. E	K	t	Max. E	L2. E	K	t	Max. E	L2. E
512	FSGS	159001	186.89	1.6899e-2	1.2372e-2	30631	42.61	1.0597e-2	7.7651e-3	4220	5.1500	4.0014e-5	2.9478e-5
	FSKSOR	1400	4.4000	1.6901e-2	1.2373e-2	524	1.8999	1.0599e-2	7.7666e-3	335	0.5699	4.1779e-5	3.0745e-5
	HSKSOR	702	0.5899	1.6855e-2	1.2339e-2	270	0.2500	1.0446e-2	7.6541e-3	171	0.3300	3.6599e-4	1.7873e-4
1024	FSGS	564746	1670.09	1.6892e-2	1.2367e-2	110487	525.92	1.0591e-2	7.7604e-3	15400	36.49	3.4572e-5	2.5555e-5
	FSKSOR	2782	21.9799	1.6901e-2	1.2373e-2	1026	6.3199	1.0600e-2	7.7666e-3	656	4.0500	4.1742e-5	3.0717e-5
	HSKSOR	1400	2.3099	1.6855e-2	1.2339e-2	524	0.9199	1.0446e-2	7.6539e-3	335	0.6299	3.6605e-4	2.5279e-4
2048	FSGS	1973904	10396.34	1.6863e-2	1.2347e-2	393817	2314.57	1.0565e-2	7.6539e-3	55762	262.80	1.2772e-5	9.7891e-6
	FSKSOR	5495	60.3999	1.6901e-2	1.2373e-2	2152	23.7200	1.0599e-2	7.7666e-3	1286	14.4700	4.1649e-5	3.0649e-5
	HSKSOR	2782	16.1299	1.6855e-2	1.2339e-2	1026	3.4499	1.0446e-2	7.7666e-3	656	2.2500	3.6605e-4	2.5280e-4
4096	FSGS	6899203	34717.02	1.6835e-2	1.2323e-2	1403713	7185.97	1.0523e-2	7.7261e-3	199761	2102.79	7.4995e-5	5.3732e-5
	FSKSOR	10872	231.8199	1.6901e-2	1.2373e-2	4221	89.5699	1.0600e-2	7.7667e-3	2518	55.7500	4.1469e-5	3.0519e-5
	HSKSOR	5495	58.1299	1.6855e-2	1.2339e-2	2152	14.5800	1.0446e-2	7.6539e-3	1286	8.7199	3.6612e-4	2.5287e-4
8192	FSGS	24114142	82862.28	1.6832e-2	1.2278e-2	30631	42.61	1.0486e-2	7.6157e-3	705935	14915.22	4.2607e-4	3.0859e-4
	FSKSOR	22196	511.1800	1.6901e-2	1.2373e-2	524	1.8999	1.0600e-2	7.7667e-3	4929	120.44	4.0877e-5	3.0096e-5
	HSKSOR	10871	127.7599	1.6855e-2	1.2339e-2	270	0.2500	1.0446e-2	7.6539e-3	2518	35.7099	3.6626e-4	2.5299e-4

Table 2.
The computed numerical result for Example 2 at α=0.333,0.666 and 0.999.

M	Method	$\alpha = 0.333$				$\alpha = 0.666$				$\alpha = 0.999$			
		K	t	Max. E	L2. E	K	t	Max. E	L2. E	K	t	Max. E	L2. E
512	FSGS	174517	425.64	1.3047e-1	9.5039e-2	31746	55.89	8.1039e-2	5.9083e-2	4278	7.75	4.1287e-4	3.0292e-4
	FSKSOR	1446	2.2299	1.3048e-1	9.5041e-2	539	0.8999	8.1042e-2	5.9090e-2	333	0.5100	4.1471e-4	3.0424e-4
	HSKSOR	724	0.5900	1.3045e-1	9.5021e-2	278	0.2799	8.0928e-2	5.9008e-2	170	0.1600	1.7599e-4	1.2684e-4
1024	FSGS	622780	1935.86	1.3047e-1	9.5033e-2	114798	551.79	8.1032e-2	5.9083e-2	15640	46.83	4.0724e-4	2.9886e-4
	FSKSOR	2874	8.5999	1.3048e-1	9.5041e-2	1038	3.3299	8.1042e-2	5.9090e-2	652	3.9500	4.1468e-4	3.0422e-4
	HSKSOR	1446	2.2200	1.3045e-1	9.5017e-2	539	0.9599	8.0924e-2	5.9005e-2	333	0.5500	1.7606e-4	1.2711e-4
2048	FSGS	2189862	14451.59	1.3043e-1	9.5010e-2	410449	2864.44	8.1000e-2	5.9061e-2	56743	99.42	3.8475e-4	2.8259e-4
	FSKSOR	5704	35.6599	1.3048e-1	9.5041e-2	2062	12.9799	8.1042e-2	5.9090e-2	1278	13.7300	4.1457e-4	3.0414e-4
	HSKSOR	2875	8.5900	1.3044e-1	9.5016e-2	1038	3.5399	8.0923e-2	5.9004e-2	652	2.0099	1.7603e-4	1.2715e-4
4096	FSGS	6996581	35710.75	1.3037e-1	9.4845e-2	1446737	8148.05	8.0878e-2	5.8974e-2	203775	2162.71	2.9574e-4	2.1784e-4
	FSKSOR	11250	140.4299	1.3048e-1	9.5041e-2	4193	50.8600	8.1042e-2	5.9090e-2	2503	79.3100	4.1436e-4	3.0398e-4
	HSKSOR	5704	34.7800	1.3044e-1	9.5016e-2	2063	13.6300	8.0923e-2	5.9004e-2	1278	7.6999	1.7591e-4	1.2709e-4
8192	FSGS	24637082	84184.52	1.3015e-1	9.4601e-2	5136775	45826.47	8.0652e-2	5.8762e-2	722380	16253.43	1.8953e-4	1.7853e-4
	FSKSOR	22264	523.3400	1.3048e-1	9.5041e-2	8194	191.3600	8.1042e-2	5.9090e-3	4906	204.2399	4.1366e-4	3.0347e-4

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

	HSKSOR	11250	133.5900	1.3044e-1	9.5016e-2	4193	50.1800	8.0923e-2	5.9004e-3	2503	31.2099	1.7563e-4	1.2691e-4

Table 3.
The computed numerical result for Example 3 at α=0.333,0.666 and 0.999.

M	Method	$\alpha = 0.333$				$\alpha = 0.666$				$\alpha = 0.999$			
		K	t	Max. E	L2. E	K	t	Max. E	L2. E	K	t	Max. E	L2. E
512	FSGS	166339	195.11	1.7026e-2	1.2502e-2	32817	38.59	1.0862e-2	7.9709e-3	4510	5.35	3.0790e-5	2.6575e-5
	FSKSOR	1482	2.2199	1.7028e-2	1.2504e-2	575	0.8999	1.0864e-2	7.9767e-3	357	0.5800	3.7895e-5	2.7825e-5
	HSKSOR	745	0.5899	1.6863e-2	1.2382e-3	294	0.2600	1.0318e-2	7.5783e-3	182	0.5800	1.5090e-3	7.4178e-4
1024	FSGS	597191	1407.43	1.7019e-2	1.2497e-2	119309	281.68	1.0856e-2	7.9709e-3	16554	39.29	3.0790e-5	2.2711e-5
	FSKSOR	2943	8.5500	1.7028e-2	1.2504e-2	1126	3.3600	1.0864e-2	7.9767e-3	699	2.1100	3.7864e-5	2.7800e-5
	HSKSOR	1482	2.2099	1.6862e-2	1.2381e-2	586	0.9099	1.0317e-2	7.5778e-3	357	0.5800	1.5091e-3	1.0491e-3
2048	FSGS	2116129	10971.26	1.6993e-2	1.2479e-2	429452	2054.15	1.0831e-2	7.9536e-3	60351	286.78	9.1442e-6	7.0955e-6
	FSKSOR	5841	35.1800	1.7028e-2	1.2504e-2	2250	13.4500	1.0864e-2	7.9767e-3	1372	8.0599	3.7792e-5	2.7744e-5
	HSKSOR	2943	8.6300	1.6862e-2	1.2381e-2	699	2.1500	1.5091e-3	1.0491e-3	699	2.1500	1.5091e-3	1.0491e-3
4096	FSGS	6875341	34877.02	1.6843e-2	1.2347e-2	1461457	10185.97	1.0796e-2	7.9351e-3	218015	2299.14	7.8293e-5	5.6244e-5
	FSKSOR	11612	136.7799	1.7028e-2	1.2504e-2	4476	55.6199	1.0864e-2	7.9767e-3	2690	33.3800	3.7605e-5	2.7605e-5
	HSKSOR	5841	38.8099	1.6862e-2	1.2381e-2	2250	13.8799	1.0317e-2	7.5777e-3	1372	8.0000	1.5092e-3	1.0491e-3
8192	FSGS	24585345	85662.31	1.6782e-2	1.2256e-2	1587123	43885.71	1.0751e-2	7.9063e-3	741257	17915.22	4.2625e-4	3.0945e-4
	FSKSOR	22910	532.1899	1.7028e-2	1.2503e-2	9257	216.3700	1.0864e-2	7.9767e-3	5272	125.2300	3.7193e-5	2.7302e-5
	HSKSOR	11612	144.7900	1.6862e-2	1.2381e-2	4476	62.1699	1.0317e-2	7.5776e-3	2690	35.9000	1.5093e-3	1.0493e-3

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

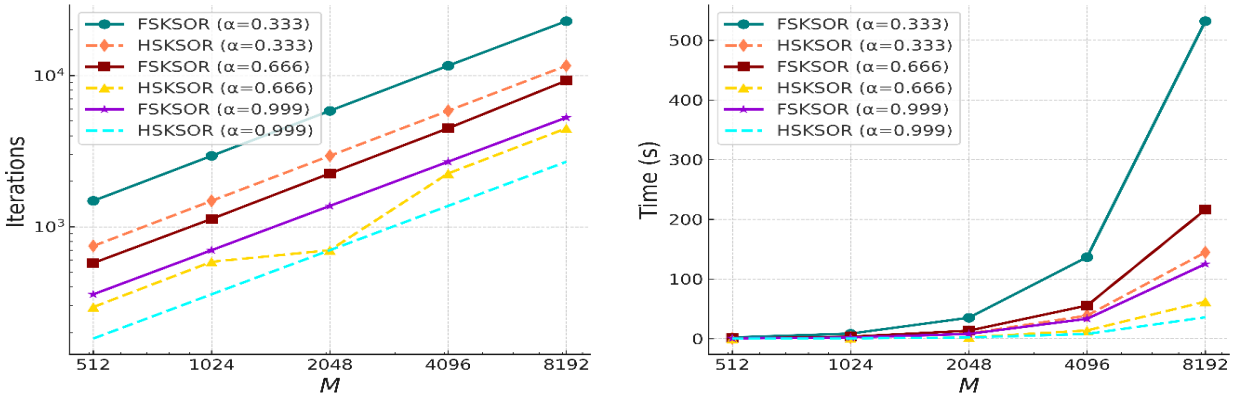


Figure 4.
Iteration count and computation time (seconds) plotted against M of two different approach based on three alpha values for Example 3

The method-maintained error rates comparable to or better than those achieved by FSGS and FSKSOR across all test cases, as verified by both maximum error and L_2-norm error assessments. This confirms that HSKSOR offers an optimal balance of computational efficiency and accuracy, making it particularly suited for handling large-scale systems and lower fractional orders. Looking ahead, the scalability and versatility of HSKSOR suggests promising avenues for future research. Extending this method to two-dimensional problems and exploring its application in parallel computing environments could further expand its utility, particularly for complex real-world simulations.

Table 4.
Reduction % of K and t between FSKSOR and HSKSOR compared to FSGS for Example 1.

M	Method	α = 0.333		α = 0.666		α = 0.999	
		K	t	K	t	K	t
512	FSKSOR	99.12%	97.65%	98.29%	95.54%	92.06%	88.93%
	HSKSOR	99.56%	99.68%	99.12%	99.41%	95.95%	93.59%
1024	FSKSOR	99.51%	98.68%	99.07%	98.79%	95.74%	88.90%
	HSKSOR	99.75%	99.86%	99.53%	99.83%	97.82%	98.27%
2048	FSKSOR	99.72%	99.42%	99.45%	98.98%	97.69%	94.49%
	HSKSOR	99.86%	99.84%	99.74%	99.85%	98.82%	99.14%
4096	FSKSOR	99.84%	99.33%	99.69%	98.75%	98.74%	97.35%
	HSKSOR	99.92%	99.83%	99.85%	99.80%	99.36%	99.59%
8192	FSKSOR	99.91%	99.38%	99.84%	99.05%	99.30%	99.19%
	HSKSOR	99.95%	99.85%	99.91%	99.87%	99.64%	99.76%

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

Table 5.
Reduction % of K and t between FSKSOR and HSKSOR compared to FSGS for Example 2

M	Method	α = 0.333		α = 0.666		α = 0.999	
		K	t	K	t	K	t
512	FSKSOR	99.17%	99.48%	98.30%	98.39%	92.22%	93.42%
	HSKSOR	99.59%	99.86%	99.12%	99.50%	96.03%	97.94%
1024	FSKSOR	99.54%	99.56%	99.09%	99.40%	95.83%	91.57%
	HSKSOR	99.77%	99.89%	99.53%	99.83%	97.88%	98.83%
2048	FSKSOR	99.74%	99.75%	99.50%	99.55%	97.75%	86.19%
	HSKSOR	99.87%	99.94%	99.75%	99.88%	98.85%	97.98%
4096	FSKSOR	99.84%	99.61%	99.71%	99.38%	98.77%	96.33%
	HSKSOR	99.92%	99.90%	99.85%	99.83%	99.37%	99.64%
8192	FSKSOR	99.91%	99.38%	99.84%	99.58%	99.32%	98.74%
	HSKSOR	99.95%	99.84%	99.92%	99.89%	99.65%	99.81%

Table 6.
Reduction % of K and t between FSKSOR and HSKSOR compared to FSGS for Example 3.

M	Method	α = 0.333		α = 0.666		α = 0.999	
		K	t	K	t	K	t
512	FSKSOR	99.11%	98.86%	98.25%	97.67%	92.08%	89.16%
	HSKSOR	99.55%	99.69%	99.10%	99.33%	95.96%	89.16%
1024	FSKSOR	99.50%	99.39%	99.06%	98.81%	95.78%	94.63%
	HSKSOR	99.75%	99.84%	99.51%	99.68%	97.84%	98.52%
2048	FSKSOR	99.72%	99.68%	99.48%	99.35%	97.73%	97.19%
	HSKSOR	99.86%	99.92%	99.84%	99.89%	98.84%	99.25%
4096	FSKSOR	99.83%	99.61%	99.70%	99.45%	98.77%	98.55%
	HSKSOR	99.92%	99.89%	99.85%	99.86%	99.37%	99.65%
8192	FSKSOR	99.91%	99.38%	99.98%	99.51%	99.30%	99.30%
	HSKSOR	99.95%	99.83%	99.99%	99.86%	99.64%	99.80%

Funding:
This research is supported by UMSGreat research grant for a postgraduate student [GUG0585-1/2023], Universiti Malaysia Sabah.

Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.

Authors' Contributions:
All authors contributed equally to the conception and design of the study. All authors have read and agreed to the published version of the manuscript.

References
[1] Q. Xu and Y. Xu, "Extremely low order time-fractional differential equation and application in combustion process," *Communications in Nonlinear Science and Numerical Simulation*, vol. 64, pp. 135-148, 2018. https://doi.org/10.1016/j.cnsns.2018.04.021

[2] S. Edeki, O. Ugbebor, and E. Owoloko, "Analytical solutions of a time-fractional nonlinear transaction-cost model for stock option valuation in an illiquid market setting driven by a relaxed Black–Scholes assumption," *Cogent Mathematics*, vol. 4, no. 1, p. 1352118, 2017. https://doi.org/10.1080/23311835.2017.1352118

[3] P. Paradisi, "Fractional calculus in statistical physics: The case of time fractional diffusion equation," *Communications in Applied and Industrial Mathematics*, vol. 6, no. 2, pp. 1-25, 2015.

[4] R. Almeida, N. R. Bastos, and M. T. T. Monteiro, "Modeling some real phenomena by fractional differential equations," *Mathematical Methods in the Applied Sciences*, vol. 39, no. 16, pp. 4846-4855, 2016. https://doi.org/10.1002/mma.3818

[5] F. J. Gaspar and C. Rodrigo, "Multigrid waveform relaxation for the time-fractional heat equation," *SIAM Journal on Scientific Computing*, vol. 39, no. 4, pp. A1201-A1224, 2017. https://doi.org/10.1137/16M1090193

[6] R. Ghaffari and F. Ghoreishi, "Reduced spline method based on a proper orthogonal decomposition technique for fractional sub-diffusion equations," *Applied Numerical Mathematics*, vol. 137, pp. 62-79, 2019. https://doi.org/10.1016/j.apnum.2018.11.014

[7] S. Kumar, *"Crank-nicolson quasi-wavelet method for the numerical solution of variable-order time-space riesz fractional reaction-diffusion equation," In Applications of Fractional Calculus to Modeling in Dynamics and Chaos*. Chapman and Hall/CRC, 2022.

[8] Y.-M. Wang and L. Ren, "A high-order L2-compact difference method for Caputo-type time-fractional sub-diffusion equations with variable coefficients," *Applied Mathematics and Computation*, vol. 342, pp. 71-93, 2019. https://doi.org/10.1016/j.amc.2018.09.007

[9] L. Wu, Y. Zhao, and X. Yang, "Alternating segment explicit-implicit and implicit-explicit parallel difference method for time fractional sub-diffusion equation," *Journal of Applied Mathematics and Physics*, vol. 6, no. 5, pp. 1017-1033, 2018. https://doi.org/10.4236/jamp.2018.65089

[10] A. Sunarto, J. Sulaiman, and A. Saudi, "Application of the full-sweep AOR iteration concept for space-fractional diffusion equation," *Journal of Physics: Conference Series*, vol. 710, no. 1, p. 012019, 2016. https://doi.org/10.1088/1742-6596/710/1/012019

[11] A. Sunarto, J. Sulaiman, and A. Saudi, "Implicit finite difference solution for time-fractional diffusion equations using AOR method," *Journal of Physics: Conference Series*, vol. 495, no. 1, p. 012032, 2014. https://doi.org/10.1088/1742-6596/495/1/012032

[12] I. Podlubny, *Fractional differential equations*. San Diego: Academic Press, 1999.

[13] W. Zahra and S. Elkholy, "Cubic spline solution of fractional Bagley-Torvik equation," *Electronic Journal of Mathematical Analysis and Applications*, vol. 1, no. 2, pp. 230-241, 2013. https://doi.org/10.21608/ejmaa.2013.309805

[14] D. Evans, "Group explicit iterative methods for solving large linear systems," *International Journal of Computer Mathematics*, vol. 17, no. 1, pp. 81-108, 1985. https://doi.org/10.1080/00207168508803452

[15] I. Youssef, "On the successive overrelaxation method," *Journal of Mathematics and Statistics*, vol. 8, no. 2, pp. 176-184, 2012. https://doi.org/10.3844/jmssp.2012.176.184

[16] N. Radzuan, M. Suardi, and J. Sulaiman, "KSOR iterative method with quadrature scheme for solving system of Fredholm integral equations of second kind," *Journal of Fundamental and Applied Sciences*, vol. 9, no. 5S, pp. 609-623, 2017. https://doi.org/10.4314/jfas.v9i5s.43

[17] M. U. Alibubin, A. Sunarto, and J. Sulaiman, "Quarter-sweep nonlocal discretization scheme with QSSOR iteration for nonlinear two-point boundary value problems," *In Journal of Physics: Conference Series*, no. 1, p. 012023, 2016. https://doi.org/10.1088/1742-6596/710/1/012023

[18] D. M. Young, *Iterative solution of large linear systems*. Elsevier, 2014.

[19] D. M. Young, "Second-degree iterative methods for the solution of large linear systems," *Journal of Approximation Theory*, vol. 5, no. 2, pp. 137-148, 1972. https://doi.org/10.1016/0021-9045(72)90036-6

[20] I. K. Youssef and A. Taha, "On the modified successive overrelaxation method," *Applied Mathematics and Computation*, vol. 219, no. 9, pp. 4601-4613, 2013. https://doi.org/10.1016/j.amc.2012.10.071

[21] B. Baharuddin, A. Sunarto, and J. Dale, "QSAOR iterative method for the solution of time fractional diffusions equation," *Journal Iteratif of Enginering and Applied Science*, vol. 12, no. 12, pp. 3220-3224, 2017. https://doi.org/10.36478/jeasci.2017.3220.3224

[22] A. Sunarto, J. Sulaiman, and A. Saudi, "Implementation of the half-sweep AOR iterative algorithm for space-fractional diffusion equations," *Jurnal Teknologi*, vol. 78, no. 6-4, pp. 7-12, 2016. https://doi.org/10.11113/jt.v78.8968

[23] A. Demir, S. Erman, B. Özgür, and E. Korkmaz, "Analysis of fractional partial differential equations by Taylor series expansion," *Boundary Value Problems*, vol. 2013, pp. 1-12, 2013. https://doi.org/10.1186/1687-2770-2013-68

[24] N. F. A. Zainal, J. Sulaiman, and M. U. Alibubin, "Application of four-point EGSOR iteration with nonlocal arithmetic mean discretization scheme for solving Burger's equation," in *Conference Proceedings, AIP Publishing*, 2018, vol. 2013, no. 1, doi: https://doi.org/10.1063/1.5054234.

[25] N. F. A. Zainal, J. Sulaiman, and M. U. Alibubin, "Application of SOR iteration with nonlocal arithmetic discretization scheme for solving Burger's equation," *Journal of Physics: Conference Series*, vol. 1358, p. 012051, 2019. https://doi.org/10.1088/1742-6596/1358/1/012051

[26] Y. Zhang, "A finite difference method for fractional partial differential equation," *Applied Mathematics and Computation*, vol. 215, no. 2, pp. 524-529, 2009. https://doi.org/10.1016/j.amc.2009.05.018

[27] F. A. Muhiddin, J. Sulaiman, and A. Sunarto, "Grünwald implicit solution for solving one-dimensional time-fractional parabolic equations using SOR iteration," *Journal of Physics: Conference Series*, vol. 1358, no. 1, p. 012055, 2019. https://doi.org/10.1088/1742-6596/1358/1/012055

[28] S. B. Yuste, "Weighted average finite difference methods for fractional diffusion equations," *Journal of Computational Physics*, vol. 216, no. 1, pp. 264-274, 2006. https://doi.org/10.1016/j.jcp.2005.12.006

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 9, No. 1: 896-911, 2025
DOI: 10.55214/25768484.v9i1.4269

[29] N. J. Ford, J. Xiao, and Y. Yan, "A finite element method for time fractional partial differential equations," *Fractional Calculus and Applied Analysis*, vol. 14, pp. 454-474, 2011. https://doi.org/10.2478/s13540-011-0028-2

[30] R. Rahman, N. A. M. Ali, J. Sulaiman, and F. A. Muhiddin, "Caputo's finite difference solution of fractional two-point BVPs using AGE iteration," *Journal of Physics: Conference Series*, vol. 1123, no. 1, p. 012044, 2018. https://doi.org/10.1088/1742-6596/1123/1/012044

[31] R. Li, L. Gong, and M. Xu, "A heterogeneous parallel Red–Black SOR technique and the numerical study on SIMPLE," *The Journal of Supercomputing*, vol. 76, pp. 9585-9608, 2020. https://doi.org/10.1007/s11227-020-03221-1

[32] K. Wang and H. Wang, "A fast characteristic finite difference method for fractional advection–diffusion equations," *Advances in Water Resourses*, vol. 34, no. 7, pp. 810-816, 2011. https://doi.org/10.1016/j.advwatres.2010.11.003

[33] M. U. Alibubin, J. Sulaiman, F. A. Muhiddin, and A. Sunarto, "Implementation of the KSOR method for solving one-dimensional time-fractional parabolic partial differential equations with the caputo finite difference scheme title of manuscript," *Journal of Advanced Research in Applied Sciences and Engineering Technology*, vol. 48, no. 1, pp. 168-179, 2024. https://doi.org/10.37934/araset.48.1.168179

[34] D. A. Murio, "Implicit finite difference approximation for time fractional diffusion equations," *Computers & Mathematics with Applications*, vol. 56, no. 4, pp. 1138-1145, 2008. https://doi.org/10.1016/j.camwa.2008.02.015

[35] A. R. F. Sabdin, C. H. C. Hussin, A. Mandangan, and J. Sulaiman, "An efficient semi-analytical method by using adaptive approach in solving nonlinear schrödinger equations," *Semarak International Journal of Fundamental and Applied Mathematics*, vol. 2, no. 1, pp. 1-12, 2024. https://doi.org/10.1234/semarak.2024.001

[36] N. Kamarudin *et al.*, "Unveiling collaborative trends in fuzzy delphi method (FDM) research: A co-authorship bibliometrics study," *International Journal of Computational Thinking and Data Science*, vol. 2, no. 1, pp. 1-20, 2024. https://doi.org/10.1234/semarak.2024.001

[37] M. U. Alibubin, J. Sulaiman, and A. Sunarto, "Performance analysis of half-sweep AOR method with nonlocal discretization scheme for nonlinear two-point boundary value problem," *American Institute of Physics*, vol. 1739, no. 1, pp. 020068-1– 020068-7, 2016. https://doi.org/10.1063/1.4952548

[38] I. Karatay, Ş. R. Bayramoğlu, and A. Şahin, "Implicit difference approximation for the time fractional heat equation with the nonlocal condition," *Applied Numerical Mathematics*, vol. 61, no. 12, pp. 1281-1288, 2011. https://doi.org/10.1016/j.apnum.2011.08.007