Mathematical and Computational Applications

MDPI

Article

A Novel Fractional-Order Scheme for Non-Linear Problems with Applications in Optimization

Mudassir Shams [1,2], Nasreen Kausar [1,3,4] and Pourya Pourhejazy [5,*]

[1] Department of Mathematics, Faculty of Arts and Science, Balikesir University, Balikesir 10145, Turkey; mudassir.shams@balikesir.edu.tr (M.S.); kausar.nasreen57@gmail.com (N.K.)
[2] Department of Mathematics and Statistics, Riphah International University, Islamabad 44000, Pakistan
[3] Department of Mathematics, Faculty of Engineering and Natural Sciences, Istinye University, Sarıyer, Istanbul 34396, Turkey
[4] Department of Mathematics, Faculty of Arts and Science, Yildiz Technical University, Esenler, Istanbul 34220, Turkey
[5] Department of Industrial Engineering, UiT The Arctic University of Norway, Lodve Langesgate 2, 8514 Narvik, Norway
* Correspondence: pourya.pourhejazy@uit.no

Abstract

The existing methods for solving non-linear equations encounter convergence issues and computing constraints, especially when used in fractional-order or complex non-linear problems. This study develops a higher-order fractional technique for solving non-linear equations based on the Caputo fractional derivative. The proposed method uses a fractional framework to improve local convergence and stability while ensuring high efficiency in every iteration step. Local convergence analysis using generalized Taylor series expansion reveals that the order of the new fractional scheme for solving non-linear equations is $5\wp + 1$, where $\wp \in (0,1]$ represents the Caputo fractional order, determining the memory depth of the Caputo fractional derivative. The performance of the method is further investigated using a variety of non-linear problems from engineering optimization and applied sciences, such as engineering control systems, computational chemistry, thermodynamics models, and operations research, such as inventory optimization. Analyzing the key performance metrics, such as dynamical analysis, percentage convergence, residual error, and computation time, confirms the advantages of the developed method over the state-of-the-art. This study provides a solid framework for higher-order fractional iterative approaches, paving the way for advanced applications of non-linear problems.

Keywords: convergence; residual error graph; CPU-time; computational convergence order; optimization

Academic Editors: Renhai Wang and Junesang Choi

Received: 15 December 2025
Revised: 11 February 2026
Accepted: 24 February 2026
Published: 3 March 2026

1. Introduction

Fractional non-linear equations can simulate complex systems characterized by memory and hereditary characteristics [1–3]. Compared to classical calculus, fractional non-linear equations incorporate fractional derivatives with greater flexibility and precision in capturing real-world dynamics. This type of equation is commonly used when the state of material/systems is determined by its prior histories, for example in inventory management optimization models, signal processing [4], biological models [5,6], and viscoelasticity [7], among others.

Fractional models are widely applied in engineering science to establish robust controllers, optimize heat and mass transfer processes, and model energy storage systems [8].

In natural sciences (e.g., physics), fractional models are employed to describe anomalous diffusion [9] and non-local events in heterogeneous media [10]. Fractional non-linear equations play a significant role in biology and medicine, where they are employed to model neuronal dynamics, drug delivery, tissue mechanics, and epidemic processes [11,12]. In addition, Caputo-based fractional diffusion schemes have proven effective in image processing and inverse problems, particularly for reaction–diffusion modeling and edge-preserving regularization [13,14]. Recent studies further extend these approaches to non-linear parabolic PDEs with variable growth structures for multi-frame MRI reconstruction [15].

The trans-disciplinary use cases of fractional models span from financial modeling to seismic analysis, where conventional methods cannot account for long-term interdependence. The versatility and precision of fractional models [16] make them suitable for addressing complex problems in engineering science and technology [17–19].

The analytical approaches to solve non-linear fractional-order equations are often infeasible for real-world applications due to the following limitations:

- These approaches are quite complex and may not necessarily result in closed-form solutions.
- Non-linearity further complicates the identification of exact solutions with fractional derivatives, resulting in new obstacles that do not exist in integer-order systems.
- Obtaining analytical solutions can be time-intensive, especially for equations involving fractional operators with Caputo and Riemann–Liouville derivatives, among other methods.
- Fractional derivatives bring with them memory issues, making traditional locality-based analytical procedures inapplicable.
- Analytical methods are inefficient for tackling multiple-dimensional, large-scale, and strongly linked systems.

In contrast, numerical techniques provide a more versatile and effective means for solving non-linear equations

$$f(x) = 0. \tag{1}$$

They enable approximation to a broad class of optimization problems characterized by high non-linearity and real-world-oriented boundary conditions. Numerical techniques can be adjusted for various fractional operators. Taking Caputo as an example, numerical techniques are highly effective in managing large-scale systems, increasing computational efficiency, allowing for stability and convergence analysis, and enabling adaptive tactics to improve accuracy. Most importantly, solutions obtained by numerical techniques can be displayed and investigated for a deeper understanding of the dynamics in the system, which is not applicable with pure analytical approaches. Among computational schemes, classical and traditional fractional-order techniques are likely to have intrinsic flaws when dealing with non-linear problems, especially those involving memory-dependent or multi-dimensional dynamics. Key drawbacks include:

- Problems involving memory effects or singularities have low precision.
- They are often characterized by low convergence rates, especially for stiff or highly non-linear systems.
- In most cases, iterative techniques diverge if the function's derivative gets close to zero around the starting point.
- The local convergence behavior of iterative techniques may potentially result in divergence or inaccurate outcomes for non-linear complex fractional problems.
- They are sensitive to initial assumptions, resulting in divergence or convergence to undesirable roots.
- In practical applications, long-range dependencies are not captured.

https://doi.org/10.3390/mca31020040

Several fractional-order approaches [20–22] have been developed, most of which suffer from delayed convergence or limited stability zones [23] in complicated non-linear systems.

Recent developments in fractional-order numerical methods aimed to achieve a convergence order of up to $2\mathfrak{c}+1$, where $\mathfrak{c}$ is the fractional parameter determining the differentiation order. Several existing methods are converted into fractional-order numerical schemes utilizing fractional derivatives to achieve a better convergence order; Refs. [24–26] are the seminal examples. Although these methods improve the flexibility and precision of the solution procedure for non-linear problems, the highest convergence order remains $2\mathfrak{c}+1$. This is mostly due to the limitations of the iterative techniques that are meant to balance high accuracy, processing costs, and stability issues inherent in fractional calculus.

This study develops a fractional-order numerical method based on the Caputo fractional derivative $\left(\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right]\right)$ to overcome the mentioned limitations. The developed method is projected to enhance convergence orders to $5\mathfrak{c}+1$, which is better than the existing state-of-the-art. The method is also expected to improve both the efficiency and precision of solving non-linear equations. This development has implications for application areas where precise optimization solutions are expected in a short computational time. The contributions of this study are summarized as follows.

- A fractional-order method of the highest convergence order is proposed. The performance of the method is examined against the existing methods for solving non-linear problems in operations research and engineering optimization.
- Generalised Taylor series are explored to conduct a local theoretical convergence analysis, proving that the fractional approach has a convergence order of $5\mathfrak{c}+1$, where $\mathfrak{c} \in (0,1]$. A fractal representation of the proposed technique is further developed, contributing to a better understanding of its structure and allowing for implementation in more complex application areas.
- The stability and robustness of the suggested scheme are investigated under various conditions and considering the percentage zones of convergence and divergence. A rigorous theoretical convergence order is targeted, based on which, a theoretical foundation for efficiency and reliability is established. Efficient method execution, which lowers iterations, is helpful for immense and memory-constrained problems. Numerous comparisons reveal that the proposed method outperforms the state-of-the-art in terms of accuracy, convergence rate, computational time in seconds, and percentage convergence.
- The applicability of the proposed approach is demonstrated using four numerical examples in operations research and engineering optimization. Findings are compared to existing numerical methods, considering account precision, stability, and computational efficiency.

Within the comparable classes of methods, the proposed scheme extends a high-order classical multi-step framework to the fractional setting via Caputo derivatives and, to the best of our knowledge, represents one of the few fractional Newton-type methods that jointly achieve multi-step correction and high local convergence order with a balanced computational cost for single-root problems.

The remainder of the article is organized as follows. Section 2 proposes and examines the fractional-order scheme. Section 3 analyzes and discusses the basins of attraction to find a zone of percentage convergence. On this basis, the visual and numerical perception of the methods' convergence attributes is provided. Section 4 investigates the engineering optimization and operations research applications and challenges to evaluate the consistency, efficiency, and stability of the newly developed method, comparing it with the existing methods. Finally, Section 5 concludes the study and provides suggestions for extending the findings.

2. Development and Analysis of Fractional Iterative Schemes

The numerical methods based on fractional order offer fair approximations when analytical solutions are impractical. They help capture complex real-world dynamics, thanks to the ability to handle memory effects and non-local behaviors inherent in fractional systems. They are appropriate for solving large-scale, multidimensional problems and are flexible to adjust to different fractional operators and boundary conditions. The computational efficiency and capacity of the numerical methods based on fractional order offer are essential to studying stability and convergence. They connect theoretical developments with real-world applications by facilitating the modeling of complex phenomena. Compared to classical algorithms, the fractional-order iterative method offers faster and more accurate convergence, making it a reliable tool for solving non-linear equations. This section begins by presenting the fundamental concepts and preliminary definitions required for the formulation and analysis of the proposed method.

Among the commonly used fractional derivatives, such as the Riemann–Liouville, Grünwald–Letnikov, Atangana–Baleanu, and Caputo–Fabrizio derivatives, most do not satisfy the classical calculus requirement that the fractional derivative of a constant vanishes. In contrast, the Caputo fractional derivative is one of the few fractional operators that assigns a zero derivative to constant functions, thereby preserving a fundamental property of classical calculus and making it particularly suitable for modeling physical and engineering problems with standard initial conditions [27].

Definition 1 (Caputo fractional derivative)**.** *Let $f : \mathbb{R} \to \mathbb{R}$ be a function such that $f \in C^m([\xi_1, x])$, where $-\infty < \xi_1 < x < +\infty$, $\xi_1 \geq 0$, and $m = \lceil \mathfrak{c} \rceil$ with $\mathfrak{c} > 0$. The Caputo fractional derivative of order $\mathfrak{c}$ of the function f is defined by [28–30]*

$$\left[\eth_{\xi_1}^{\mathfrak{c}}\right] f(x) = \begin{cases} \dfrac{1}{\Gamma(m-\mathfrak{c})} \displaystyle\int_{\xi_1}^{x} \frac{f^{(m)}(t)}{(x-t)^{\mathfrak{c}-m+1}}\, dt, & m-1 < \mathfrak{c} < m, \\ \dfrac{d^m}{dx^m} f(x), & \mathfrak{c} = m \in \mathbb{N}, \end{cases} \tag{2}$$

where $\Gamma(\cdot)$ denotes the Gamma function defined as

$$\Gamma(x) = \int_0^{\infty} u^{x-1} e^{-u}\, du, \qquad x > 0. \tag{3}$$

The Taylor series Theorem A1 (Appendix A) provides a local approximation of a function near its root. This theorem is fundamental in developing iterative methods to solve non-linear equations. Expanding the function to the Taylor series and higher-order terms helps establish more precise and effective iterative schemes. This approximation approach serves as a basis for iterative schemes with function linearization and an acceleration to the root. Using the classical Newton's method's Caputo-type fractional version (KFM$_1^{[*]}$), Candelario et al. introduced the following variation [31]:

$$y^{[\kappa]} = x^{[\kappa]} - \left(\Gamma(\mathfrak{c}+1) \frac{f(x^{[\kappa]})}{\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(x^{[\kappa]})} \right)^{1/\mathfrak{c}}, \tag{4}$$

where $\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(x_i) \approx \left[\eth_{\xi}^{\mathfrak{c}}\right] f(\xi)$ for any $\mathfrak{c} \in \mathbb{R}$. The order of convergence of (4) is $\mathfrak{c}+1$, and error is given as:

$$e^{[*]} = \left[\frac{\Gamma(\mathfrak{c}+1) - \Gamma(\mathfrak{c}+1)}{\Gamma(\mathfrak{c}+1)} b_2 \left(e^{[\kappa]}\right)^{\mathfrak{c}+1} + O\left(\left(e^{[\kappa]}\right)^{2\mathfrak{c}+1} \right) \right], \tag{5}$$

where

$$e^{[*]} = y^{[\kappa]} - \xi,\ e^{[\kappa]} = x^{[\kappa]} - \xi$$

and

$$b_\gamma = \frac{\Gamma(\mathscr{c}+1)}{\Gamma(\gamma\mathscr{c}+1)} \frac{\left[\partial_\xi^{\gamma\mathscr{c}}\right] f(\xi)}{\left[\partial_\xi^{\mathscr{c}}\right] f(\xi)}, \gamma \geq 2.$$

A one-step fractional iterative method of convergence order $\mathscr{c}+1$ (KFM$_2^{[*]}$) was proposed by Shams et al. [32] as:

$$y^{[\kappa]} = x^{[\kappa]} - \left(\left(\Gamma(\mathscr{c}+1) \frac{f\left(x^{[\kappa]}\right)}{\left[\partial_{\mathscr{c}_1}^{\mathscr{c}}\right] f(x^{[\kappa]})} \right) \left[\frac{1}{1 - \alpha \frac{f(x^{[\kappa]})}{1+f(x^{[\kappa]})}} \right] \right)^{1/\mathscr{c}}, \tag{6}$$

and satisfies

$$e^{[*]} = \frac{\left((\alpha+b_2)\Gamma^2(\mathscr{c}+1) - b_2\Gamma(2\mathscr{c}+1)\right)}{\mathscr{c}\Gamma(\mathscr{c}+1)} b_2 \left(e^{[\kappa]}\right)^{\mathscr{c}+1} + O\left(\left(e^{[\kappa]}\right)^{2\mathscr{c}+1}\right). \tag{7}$$

Ali et al. [33] proposed the following scheme (KFM$_3^{[*]}$) of convergence order as

$$z^{[\kappa]} = y^{[\kappa]} - \Gamma(\mathscr{c}+1) \frac{f\left(y^{[\kappa]}\right)}{\left[\partial_{\mathscr{c}_1}^{\mathscr{c}}\right] f(x^{[\kappa]}) - \phi f\left(y^{[\kappa]}\right)}, \tag{8}$$

where

$$y^{[\kappa]} = x^{[\kappa]} - \Gamma(\mathscr{c}+1) \frac{f\left(x^{[\kappa]}\right)}{\left[\partial_{\mathscr{c}_1}^{\mathscr{c}}\right] f(x^{[\kappa]}) - \phi f\left(x^{[\kappa]}\right)},$$

$\phi \in \mathbb{R}$ as well as satisfying the subsequent error equations:

$$e^{[*]} = \left(\frac{b_2^{\mathscr{c}}}{\Gamma(\mathscr{c}+1)} \right) \left(e^{[\kappa]}\right)^{2\mathscr{c}^2} + \frac{\left(\frac{\phi}{\Gamma(\mathscr{c}+1)}\right)^{\mathscr{c}}}{\Gamma(\mathscr{c}+1)} + O\left(\left(e^{[\kappa]}\right)^{3\mathscr{c}^2}\right). \tag{9}$$

In [34], a two-step fractional Newton type approach (KFM$_4^{[*]}$) with convergence order $2\mathscr{c}+1$ was presented as:

$$v^{[\kappa]} = y^{[\kappa]} - \left(\Gamma(\mathscr{c}+1) \frac{f(y^{[\kappa]})}{\left[\partial_{\mathscr{c}_1}^{\mathscr{c}}\right] f(x^{[\kappa]})} \right)^{1/\mathscr{c}}, \tag{10}$$

where $y^{[\kappa]} = x^{[\kappa]} - \left(\Gamma(\mathscr{c}+1) \frac{f(x^{[\kappa]})}{\left[\partial_{\mathscr{c}_1}^{\mathscr{c}}\right] f(x^{[\kappa]})} \right)^{1/\mathscr{c}}$, satisfying the subsequent error equations:

$$e^{[*]} = \frac{\Gamma(\mathscr{c}+1) - \Gamma(\mathscr{c}+1)}{\Gamma(\mathscr{c}+1)} b_2 \left(e^{[\kappa]}\right)^{2\mathscr{c}+1} + O\left(\left(e^{[\kappa]}\right)^{3\mathscr{c}+1}\right). \tag{11}$$

Remark 1. *The fractional schemes* KFM$_1^{[*]}$–KFM$_4^{[*]}$ *reviewed above belong to the class of one- and two-step Caputo-based Newton-type methods, where the fractional derivative is evaluated either at the current iterate or reused from a previous stage. In contrast to these approaches, the proposed method extends the multi-step correction strategy originally developed for classical Newton-type solvers by explicitly incorporating fractional-order memory through the Caputo operator. This results in a structurally distinct update mechanism that combines multi-evaluation correction with non-local fractional dynamics, which is not present in existing fractional Newton-type schemes.*

The method outlined in [35], which was shown to provide higher accuracy, faster convergence, robustness, and processing efficiency than existing single root-finding techniques, is the basis of the present study:

$$v^{[\kappa]} = z^{[\kappa]} - \frac{f(z^{[\kappa]}) + f(y^{[\kappa]})}{f'(y^{[\kappa]})}, \tag{12}$$

where $z^{[\kappa]} = y^{[\kappa]} - \frac{f(y^{[\kappa]})}{f'(y^{[\kappa]})}, y^{[\kappa]} = x^{[\kappa]} - \frac{f(x^{[\kappa]})}{f'(x^{[\kappa]})}$. The technique exhibits sixth-order convergence, showing significant accuracy improvement at each iterative step. The technique's local error is provided below, confirming its efficiency in approximating the root.

$$e_i^{[*]} = \left(b_4^{[*]}\left(b_2^{[*]}\right)^3 + 5\left(b_2^{[*]}\right)^3 - b_3^{[*]}b_2^{[*]}\right)\left(e^{[\kappa]}\right)^4 + O\left(\left(e^{[\kappa]}\right)^5\right), \tag{13}$$

where

$$e_i^{[*]} = x^{[\kappa]} - \xi, e^{[\kappa]} = x^{[\kappa]} - \xi$$

and

$$b_j^{[*]} = \frac{1}{j!}\frac{f^j(\xi)}{f'(\xi)}, j \geq 2.$$

From classical to fractional framework. The sixth-order method in [35] serves as the classical foundation of the present study. Rather than directly competing with existing fractional Newton-type methods, we extend this high-order classical scheme by embedding Caputo fractional derivatives to account for memory and non-local effects. This extension preserves the favorable convergence properties of the original method while enabling its application to fractional-order models, which constitutes the main contribution of this work.

The memory- and non-local-effect approach (abbreviated as $\mathrm{FSM}_1^{[**]}$) are implemented to Caputo-fractional derivatives to provide more realistic simulations:

$$v^{[\kappa]} = z^{[\kappa]} - \left(\Gamma(\mathfrak{c}+1)\frac{f(z^{[\kappa]}) + f(y^{[\kappa]})}{\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right]f(y^{[\kappa]})}\right)^{1/\mathfrak{c}}, \tag{14}$$

where $z^{[\kappa]} = y^{[\kappa]} - \left(\Gamma(\mathfrak{c}+1)\frac{f(y^{[\kappa]})}{\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right]f(y^{[\kappa]})}\right)^{1/\mathfrak{c}}$, and $y^{[\kappa]} = x^{[\kappa]} - \left(\Gamma(\mathfrak{c}+1)\frac{f(x^{[\kappa]})}{\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right]f(x^{[\kappa]})}\right)^{1/\mathfrak{c}}$.

Convergence Analysis

Convergence analysis is required for numerical systems that solve non-linear equations to ensure reliability and efficiency. This is done by determining the conditions under which the approach converges to the solution. Additionally, convergence analysis can be considered a basis for evaluating the methods and adjusting them to address specific problem types or restrictions. The local convergence of the developed method is discussed in Theorem 1.

Theorem 1. *Let $f : \eth \subseteq \mathbb{R} \to \mathbb{R}$ be a function that is continuous, with $\left[\eth_{\mathfrak{c}_1}^{m\mathfrak{c}}\right]f(x)$ of order m for any $\mathfrak{c} \geq 0$ and $\mathfrak{c} \in (0,1]$ with the exact root ξ of (1). Moreover, if the initial value $x^{[0]}$ is close enough, then (14)*

$$v^{[\kappa]} = z^{[\kappa]} - \left(\Gamma(\mathfrak{c}+1)\frac{f(z^{[\kappa]}) + f(y^{[\kappa]})}{\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right]f(y^{[\kappa]})}\right)^{1/\mathfrak{c}}, \tag{15}$$

convergence order is least $5\mathscr{c}+1$, *with the following error:*

$$v^{[\kappa]} = \left(\; g_1^{[*]} + g_2^{[*]} g_3^{[*]} + g_4^{[*]} + g_5^{[*]} + g_6^{[*]} + g_7^{[*]} \; \right) \left[e^{[\kappa]}\right]^{5\mathscr{c}+1}) + O(\left[e^{[\kappa]}\right]^{6\mathscr{c}+1}), \tag{16}$$

where $g_1^{[*]} - g_7^{[*]}$ *are shown in Appendix B.*

Proof. Let the simple solution ξ of f and $x^{[\kappa]} = \xi + e^{[\kappa]}$. Utilizing Taylor series expansion of $f(x^{[\kappa]})$ and $\left[\eth_{\mathscr{c}_1}^{\mathscr{c}}\right] f(x^{[\kappa]})$ around $x = \xi$. Taking $f(\xi) = 0$, we obtain:

$$f(x^{[\kappa]}) = \frac{\left[\eth_{\xi}^{\mathscr{c}}\right] f(\xi)}{\Gamma(\mathscr{c}+1)} \left[\left[e^{[\kappa]}\right]^{\mathscr{c}} + b_2 \left[e^{[\kappa]}\right]^{2\mathscr{c}} + b_3 \left[e^{[\kappa]}\right]^{3\mathscr{c}} \right] + O\left(\left[e^{[\kappa]}\right]^{4\mathscr{c}} \right). \tag{17}$$

Taking the fractional derivative of (17), we obtain

$$\left[\eth_{\mathscr{c}_1}^{\mathscr{c}}\right] f(x^{[\kappa]}) = \frac{\left[\eth_{\xi}^{\mathscr{c}}\right] f(\xi)}{\Gamma(\mathscr{c}+1)} \left[\; \Gamma(\mathscr{c}+1) + \tfrac{\Gamma(2\mathscr{c}+1)}{\Gamma(\mathscr{c}+1)} b_2 \left[e^{[\kappa]}\right]^{\mathscr{c}} + \tfrac{\Gamma(3\mathscr{c}+1)}{\Gamma(2\mathscr{c}+1)} b_3 \left[e^{[\kappa]}\right]^{2\mathscr{c}} \; \right] + O\left(\left[e^{[\kappa]}\right]^{3\mathscr{c}} \right). \tag{18}$$

Taking the inverse of (18), we have

$$\frac{1}{\left[\eth_{\mathscr{c}_1}^{\mathscr{c}}\right] f\left(x^{[\kappa]}\right)} = \frac{\left[\eth_{\xi}^{\mathscr{c}}\right] f(\xi)}{\Gamma(\mathscr{c}+1)} \left[\begin{array}{c} \frac{1}{\Gamma(\mathscr{c}+1)} - \frac{\Gamma(2\mathscr{c}+1) b_2 \left[e^{[\kappa]}\right]^{\mathscr{c}}}{(\Gamma(\mathscr{c}+1))^3} + a_1^{[**]} + \\ a_2^{[**]} + a_3^{[**]} \end{array} \right] + O\left(\left[e^{[\kappa]}\right]^{4\mathscr{c}} \right). \tag{19}$$

Consequently, Equations (17) and (19) are multiplied to yield

$$\frac{f(x^{[\kappa]})}{\left[\eth_{\mathscr{c}_1}^{\mathscr{c}}\right] f\left(x^{[\kappa]}\right)} = \frac{\left[e^{[\kappa]}\right]^{\mathscr{c}}}{\Gamma(\mathscr{c})\mathscr{c}} + \left(a_4^{[**]}\right) \left[e^{[\kappa]}\right]^{2\mathscr{c}} + \left(a_5^{[**]}\right) \left[e^{[\kappa]}\right]^{3\mathscr{c}} + O\left(\left[e^{[\kappa]}\right]^{4\mathscr{c}} \right). \tag{20}$$

Using (20) in the first step of (14), we have

$$y^{[\kappa]} = \xi + \left(-b_2 + \frac{(2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+1/2) b_2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} \right) \left[e^{[\kappa]}\right]^{\mathscr{c}+1} + a_6^{[**]} \left[e^{[\kappa]}\right]^{2\mathscr{c}+1} + O\left(\left[e^{[\kappa]}\right]^{3\mathscr{c}+1} \right). \tag{21}$$

Expanding $f\left(y^{[\kappa]}\right)$ around $x = \xi$ using Taylor series expansion

$$f\left(y^{[\kappa]}\right) = \frac{\left[\eth_{\xi}^{\mathscr{c}}\right] f(\xi)}{\Gamma(\mathscr{c}+1)} \left[a^{\{1\}} \left[e^{[\kappa]}\right]^{\mathscr{c}+1} + -\frac{a^{\{2\}}}{2} \left[e^{[\kappa]}\right]^{2\mathscr{c}+1} \right] + O\left(\left[e^{[\kappa]}\right]^{3\mathscr{c}+1} \right). \tag{22}$$

Taking fractional derivative of (22), yield

$$\left[\eth_{\mathscr{c}_1}^{\mathscr{c}}\right] f\left(y^{[\kappa]}\right) = \frac{\left[\eth_{\xi}^{\mathscr{c}}\right] f(\xi)}{\Gamma(\mathscr{c}+1)} \left[1 + 2\, b_2 a_1^{[*]} \left[e^{[\kappa]}\right]^{\mathscr{c}+1} + 2\, b_2 a_2^{[*]} \left[e^{[\kappa]}\right]^{2\mathscr{c}+1} + a_3^{[*]} \left[e^{[\kappa]}\right]^{3\mathscr{c}+1} + \ldots \right]. \tag{23}$$

Taking the inverse of (23) yields

$$\frac{1}{\left[\eth_{\mathscr{c}_1}^{\mathscr{c}}\right] f\left(y^{[\kappa]}\right)} = \frac{\left[\eth_{\xi}^{\mathscr{c}}\right] f(\xi)}{\Gamma(\mathscr{c}+1)} \left[\begin{array}{c} 1 - 2\, a_4^{[*]} b_2 \left[e^{[\kappa]}\right]^{\mathscr{c}+1} - 2\, a_5^{[*]} b_2 \left[e^{[\kappa]}\right]^{2\mathscr{c}+1} + \\ \left(-2 a_6^{[*]}\, b_2 - 3\, a_7^{[*]}\right) \left[e^{[\kappa]}\right]^{3\mathscr{c}+1} + \ldots \end{array} \right]. \tag{24}$$

Consequently, Equations (22) and (24) are multiplied to produce

$$\frac{f\left(y^{[\kappa]}\right)}{\left[\eth_{\xi_1}^{\mathfrak{c}}\right]f\left(y^{[\kappa]}\right)}=\left[a_8^{[*]}\left[e^{[\kappa]}\right]^{\mathfrak{c}+1}+a_9^{[*]}\left[e^{[\kappa]}\right]^{2\mathfrak{c}+1}+a_{10}^{[*]}\left[e^{[\kappa]}\right]^{3\mathfrak{c}+1}\right]+O\left(\left[e^{[\kappa]}\right]^{3\mathfrak{c}+1}\right), \quad (25)$$

Using (25) in the second step of (14), we have

$$z^{[\kappa]}=\xi+a_{11}^{[*]}\left[e^{[\kappa]}\right]^{3\mathfrak{c}+1}+a_{12}^{[*]}\left[e^{[\kappa]}\right]^{4\mathfrak{c}+1}+O\left(\left[e^{[\kappa]}\right]^{5\mathfrak{c}+1}\right). \quad (26)$$

Expanding $f\left(z^{[\kappa]}\right)$ around $z=\xi$ using Taylor series expansion

$$f\left(z^{[\kappa]}\right)=\frac{\left[\eth_{\xi}^{\mathfrak{c}}\right]f(\xi)}{\Gamma(\mathfrak{c}+1)}\left[\frac{a_{16}^{[*]}{b_2}^3\left[e^{[\kappa]}\right]^{3\mathfrak{c}+1}}{(\Gamma(\mathfrak{c}))^2\mathfrak{c}^2\pi^{\frac{3}{2}}}+\frac{{b_2}^2\left[e^{[\kappa]}\right]^{4\mathfrak{c}+1}}{(\Gamma(\mathfrak{c}))^3\mathfrak{c}^3\pi^{\frac{5}{2}}(2^{\mathfrak{c}})^2}\frac{a_{13}^{[*]}}{\Gamma\left(\mathfrak{c}+\frac{1}{2}\right)}+\cdots\right]. \quad (27)$$

Adding (22) and (27), we obtained

$$f\left(y^{[\kappa]}\right)+f\left(z^{[\kappa]}\right)=\frac{\left[\eth_{\xi}^{\mathfrak{c}}\right]f(\xi)}{\Gamma(\mathfrak{c}+1)}\left[a_{14}^{[*]}\left[e^{[\kappa]}\right]^{\mathfrak{c}+1}+a_{15}^{[*]}\left[e^{[\kappa]}\right]^{2\mathfrak{c}+1}+a_{16}^{[*]}\left[e^{[\kappa]}\right]^{3\mathfrak{c}+1}+\ldots\right]. \quad (28)$$

Consequently, Equations (24) and (28) are multiplied to yield

$$\frac{f\left(y^{[\kappa]}\right)+f\left(z^{[\kappa]}\right)}{\left[\eth_{\xi_1}^{\mathfrak{c}}\right]f\left(y^{[\kappa]}\right)}=a_{17}^{[*]}\left[e^{[\kappa]}\right]^{\mathfrak{c}+1}+a_{18}^{[*]}\left[e^{[\kappa]}\right]^{2\mathfrak{c}+1}+a_{19}^{[*]}\left[e^{[\kappa]}\right]^{3\mathfrak{c}+1}+\ldots \quad (29)$$

where $a_1^{[**]}$–$a_6^{[**]}$, $a^{\{1\}}$–$a^{\{2\}}$, $a_1^{[*]}$–$a_{19}^{[*]}$ used in (19)–(29) are shown in Appendix B respectively. Using (29) in the last step of the numerical scheme (14), we have

$$v^{[\kappa]}=\left(\ g_1^{[*]}+g_2^{[*]}g_3^{[*]}+g_4^{[*]}+g_5^{[*]}+g_6^{[*]}+g_7^{[*]}\ \right)\left[e^{[\kappa]}\right]^{5\mathfrak{c}+1}+O\left(\left[e^{[\kappa]}\right]^{6\mathfrak{c}+1}\right). \quad (30)$$

The theorem is proven supported by the above findings. □

3. Factorial Representations of the Convergence Behavior

Solution algorithms are developed to obtain more efficient and precise results. Their performance can be evaluated by identifying areas of divergence or slow convergence by analyzing the basins of attraction. The basin-of-attraction analysis is employed to provide a qualitative and quantitative assessment of the global convergence behavior of the proposed and existing fractional-order iterative schemes. Applications of this type of analysis can be found in applied mathematics, physics, operations research, and engineering, where non-linear equations are prevalent. The illustration of the basins of attraction in electrical circuit analysis, fluid dynamics, chemical reaction modeling, and structural engineering are prime examples. In addition to this, visualizing the basin of attraction provides a graphical way of comparing iterative methods considering efficiency and robustness. Analyzing the basins of attraction has the following implications.

- Basins of attraction define areas where a certain iterative scheme converges to specific roots of a non-linear solution when having an initial guess. Each basin corresponds to one root. The region of convergence for an iterative method varies along a defined boundary known as a separatrix.

- Symmetry or fractal-like features in the basin indicate intricate dependencies on the initial guess.
- Irregularities may indicate the need for better iterative formulations.
- Mapping computational time over the basins of attraction helps to select initial guesses for time-sensitive applications.
- Plotting iterative time in the basin improves the selection of guesses at their roots for time-critical applications.
- Near a root, the error decreases predictably (e.g., linearly, quadratically, cubically), but this indicates the order of convergence.
- The interaction of the function's properties (e.g., steepness and local extrema) and the basin structure helps identify possible issues while solving highly non-linear equations.
- Visualizing basins helps to improve iterative methods by refining formulas, increasing convergence, and addressing divergence obstacles.

With these, basins of attraction are known to be effective analytic and diagnostic tools for evaluating iterative structures. They provide an overview of convergence zones, iteration dynamics, error behavior, and computational efficiency. These indicators are essential for solving complex non-linear equations in science and engineering. The basins of attraction is generated for the following non-linear functions to find the efficiency of the developed fractional schemes. The non-linear test function

$$f(x) = x^3 - 2x + 1, \tag{31}$$

is selected due to its several real and complex roots and its widespread use as a benchmark problem in convergence and dynamical studies. This choice allows a fair comparison with existing fractional methods and highlights differences in stability and global convergence performance. The corresponding Caputo fractional derivative of the (31) is given as

$$\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(x) = \frac{\Gamma(4)}{\Gamma(4-\mathfrak{c})} x^{3-\mathfrak{c}} - 2\frac{\Gamma(2)}{\Gamma(2-\mathfrak{c})} x^{1-\mathfrak{c}} + \frac{\Gamma(1)}{\Gamma(1-\mathfrak{c})} x^{-\mathfrak{c}}. \tag{32}$$

All basins of attraction experiments are conducted under identical numerical settings using MATLAB R2023a on a 64-bit operating system with an Intel Core i7 processor. The stopping criteria are defined as $|x^{(k+1)} - x^{(k)}| < 10^{-3}$ or upon reaching the maximum number of iterations.

Computational time in seconds (CPU-time) comparisons are performed within the same hardware and software environment to ensure fairness. Fractional derivatives are evaluated analytically using the Caputo definition in (32), thereby avoiding numerical differentiation errors. Initial guesses are sampled uniformly over the complex domain $\Omega = [-2, 2] \times [-2, 2] \subset \mathbb{C}$ using an 800×800 grid with a maximum of 20 iterations per initial point. This domain encompasses all roots of (31) and their associated convergence regions, ensuring an unbiased and comprehensive assessment of the global convergence behavior. Basins of attraction plots are generated for all fractional methods at $\alpha = 0.99$, which closely recovers the corresponding classical version of the schemes. This fixed choice of α enables a consistent comparison of the effects of memory and multi-step structures across the considered methods.

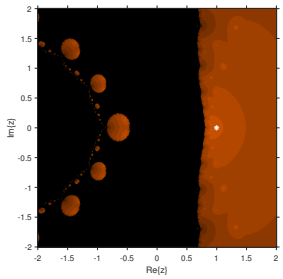
Analysis of Basins of Attraction

The basins of attraction shown in this study provide information about the suggested method's stability and robustness. Areas of attraction that are widely and evenly dispersed

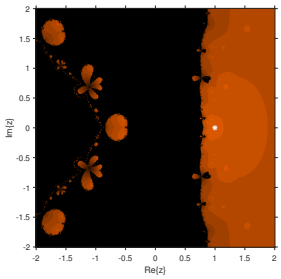
show improved tolerance and robustness for higher initial estimations. The proposed scheme performs better than alternative fractional-based techniques in

- The suggested algorithm has larger basins, indicating improved global convergence attributes.
- Symmetry within the basins ensures root stability, especially in multi-root designs.
- Smooth boundary curves in attraction areas indicate reduced sensitivity to perturbations, implying numerical stability.

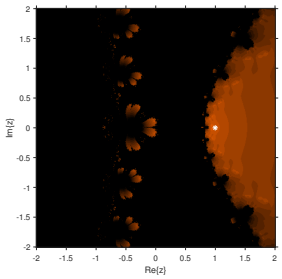
To illustrate this, the percentage of initial points converging to the exact solutions are computed and presented in Table 1. Table 1 displays the computation time in seconds, total number of points assessed (64,000), number of iterations, and percentage convergence (per-convergence). These measures allow for a general comparison of the methods' accuracy and efficiency. Table 1 provides information on each algorithm's performance in terms of speed and convergence stability. This in-depth review aids in determining the best approach to handling the provided non-linear problems. Comparing the developed method with the existing methods, it is observed that there is a larger convergence region and that the method requires fewer iterations while demonstrating consistency with the scheme's CPU-time stability for solving non-linear equations. Figure 1a–e visualizes the basins of attraction of our method compared to the existing methods. It can be observed that our a better convergence behavior and region can be obtained when compared to $KFM_1^{[*]}$–$KFM_4^{[*]}$.



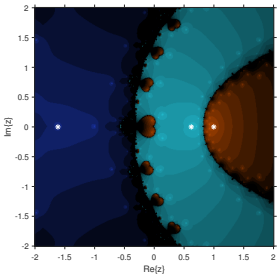
(**a**) $KFM_1^{[*]}$—Basins of attraction



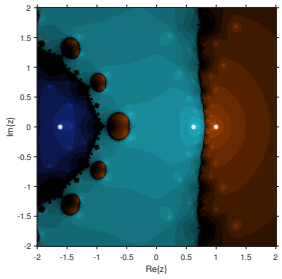
(**b**) $KFM_2^{[*]}$—Basins of attraction



(**c**) $KFM_3^{[*]}$—Basins of attraction



(**d**) $KFM_4^{[*]}$—Basins of attraction



(**e**) $FSM_1^{[**]}$—Basins of attraction

Figure 1. Basins of attraction for solving (31) using fractional Newton-type methods: (**a**) $KFM_1^{[*]}$, (**b**) $KFM_2^{[*]}$, (**c**) $KFM_3^{[*]}$, (**d**) $KFM_4^{[*]}$, and (**e**) the proposed $FSM_1^{[**]}$. Each color denotes convergence to a distinct root, while dark regions correspond to divergence or slow convergence. The proposed $FSM_1^{[**]}$ exhibits wider and smoother basins, indicating enhanced global convergence and stability.

Table 1. Quantitative performance evaluation of fractional iterative methods for solving (31).

Metrics	$\mathbf{KFM}_{\mathbf{1}}^{[*]}$	$\mathbf{KFM}_{\mathbf{2}}^{[*]}$	$\mathbf{KFM}_{\mathbf{3}}^{[*]}$	$\mathbf{KFM}_{\mathbf{4}}^{[*]}$	$\mathbf{FSM}_{\mathbf{1}}^{[**]}$
Total points	640,000	640,000	640,000	640,000	640,000
Per-convergence	54.34%	47.65%	46.65%	58.50%	86.98%
Iterations	25	24	23	24	19
CPU-time	3.454	4.343	2.3453	4.342	1.342

Remark: Per-convergence denotes the percentage of initial points that converge to a root. Total points refers to the number of mesh points used in the basin computation, and CPU-time represents the total computational time in seconds.

4. Numerical Analysis

Numerical solutions for fractional non-linear equations help validate theoretical models in practice. They provide an approximation when analytical solutions are hard or impossible to obtain. This would enable the study of complex systems with fractional dynamics, providing insight into real-world applications in physics, engineering, operations research, and finance. Furthermore, numerical findings can be used to compare and evaluate different iteration techniques to improve convergence behavior efficiency. Providing more control over modeling and support in evaluating the potential effects of fractional parameters on system behavior are among the other advantages. Given these, a numerical approach is required for solving fractional non-linear equations because obtaining exact solutions is often inapplicable. This section provides four practical examples from thermodynamics, computational chemistry, engineering control systems, and operations research.

$$e^{[\kappa]} = \left|x^{[\kappa+1]} - x^{[\kappa]}\right| < 10^{-18} \text{ and } e^{[\kappa]} = \left|f\left(x^{[\kappa]}\right)\right| < 10^{-18}, \tag{33}$$

where $e^{[\kappa]}$ absolute error. To further contextualize the numerical performance of the proposed scheme, Table 2 presents a comparative summary of representative fractional Newton-type methods for single-root problems, highlighting their structural characteristics, convergence behavior, and consistency with the classical case. All methods are assessed under the same stopping criterion given in (33) to ensure a fair and uniform comparison.

In Table 2, the column $\left[f, \left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(x)\right]$ denotes the total number of function and Caputo fractional derivative evaluations per iteration. The last column confirms that, for $\mathfrak{c} = 1$, each fractional scheme naturally reduces to its classical counterpart, with the proposed $\mathrm{FSM}_1^{[**]}$ attaining a higher classical convergence order due to its multi-step correction strategy.

Table 2. Comparison of fractional Newton-type methods for single-root problems.

Method	Steps	$\left[f, \left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(x)\right]$	Memory Effect	Convergence Order	Order at $\mathfrak{c} = 1$
$\mathrm{KFM}_1^{[*]}$ [31]	Single-step	2	No	$\mathfrak{c}+1$	2
$\mathrm{KFM}_2^{[*]}$ [32]	Single-step	2	No	$\mathfrak{c}+1$	2
$\mathrm{KFM}_3^{[*]}$ [33]	Multi-step	3	No	$2\mathfrak{c}^2$	2
$\mathrm{KFM}_4^{[*]}$ [34]	Multi-step	3	No	$2\mathfrak{c}+1$	3
$\mathrm{FSM}_1^{[**]}$ (present)	Multi-step	5	Yes	$5\mathfrak{c}+1$	6

4.1. Applications and Practical Implementation

The architecture of the new method is suitable for integration into actual software and industrial pipelines, and it has been tested in different fields, including engineering control systems, computer chemistry, and thermodynamics. The technique is modular, iterative, and computationally efficient, making it an ideal choice for integration into simulation

platforms, embedded systems, and industrial process controllers. In addition, due to its accuracy and firmness, it may enhance solvers in existing systems such as MATLAB (R2023a), Python (v3.10) (SciPy (v1.11)), and COMSOL Multiphysics (v6.1). Further work will include open-source toolkits and commercial plugin modules for process optimization, material simulation, and auto-control systems. Algorithm 1 and the flow chart in Figure 2 are used to illustrate the step-by-step procedure for implementing the new fractional schemes to solve real-world problems efficiently.

Algorithm 1 Three-step fractional-order iterative scheme for solving $f(x) = 0$

Require: Function $f(x)$, derivative $\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(x^{[\kappa]})$, initial guess $x^{[0]}$, fractional order $0 < \mathfrak{c} \leq 1$, tolerance ε, maximum iterations N

Ensure: Approximate root ξ

1: Set $\kappa \leftarrow 0$

2: **while** $\kappa < N$ and $\left|f(x^{[\kappa]})\right| > \varepsilon$ **do**

3: Compute first and second substep:

$$y^{[\kappa]} = x^{[\kappa]} - \left(\Gamma(\mathfrak{c}+1)\frac{f(x^{[\kappa]})}{\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(x^{[\kappa]})}\right)^{1/\mathfrak{c}}; z^{[\kappa]} = y^{[\kappa]} - \left(\Gamma(\mathfrak{c}+1)\frac{f(y^{[\kappa]})}{\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(y^{[\kappa]})}\right)^{1/\mathfrak{c}}.$$

4: Compute third substep:

$$v^{[\kappa]} = z^{[\kappa]} - \left(\Gamma(\mathfrak{c}+1)\frac{f(y^{[\kappa]}) + f(z^{[\kappa]})}{\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(y^{[\kappa]})}\right)^{1/\mathfrak{c}}.$$

5: $\kappa \leftarrow \kappa + 1$

6: **end while**

7: **return** $v^{[\kappa]}$

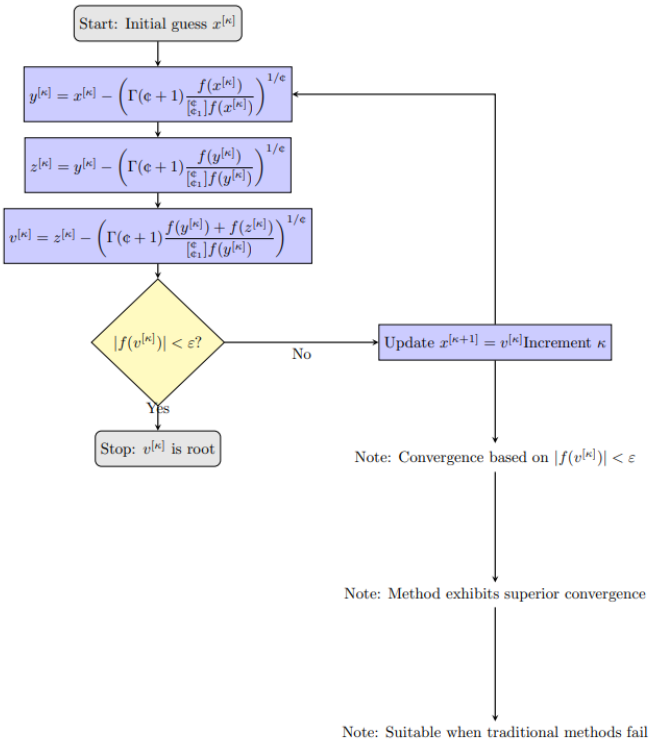


Figure 2. Flowchart of the three-step fractional order scheme for solving non-linear equations $f(x) = 0$.

Example 1. ***Memory-Based Approaches to One-Dimensional Non-linear Models in Thermodynamics and Combustion Analysis***

The adiabatic flame temperature is the temperature at which fuel burns without losing heat to its surroundings [36]. This measure determines the efficiency and energy output of a combustion system, making it a crucial parameter in thermodynamics and combustion analysis. The temperature should be determined considering the fuel type, the oxidizer type, the reactants' beginning temperatures, and the mixing ratio. Higher adiabatic flame temperatures suggest more efficient burning but also result in higher NOx emissions due to thermal dissociation. It is frequently utilized to maximize performance while developing industrial furnaces, turbines, and engines. Engineers compute it under constant pressure or volume conditions, which provides vital information about the combustion model. The following non-linear equations are raised by this notion, which is also crucial for researching the safety and environmental effects of combustion processes.

$$f(x) = \left[\frac{0.283\times10^{-6}}{3}\left(x^3 - 298^3\right) + \frac{0.283\times10^{-6}}{2}\left(x^2 - 298^2\right) + 7.256(x - 298) - 57798 \right], \quad (34)$$

where the exact root of (34) are

$$4305.30991390817, -8242.76096423642 - 8941.52674877044i.$$

The initial guess $x^{[0]} = 4305.7$ is selected through a coarse functional scan, ensuring proximity to the real root and compliance with local convergence conditions. The derivative of Equation (34) of the Caputo type is

$$\left[\eth^{\mathfrak{c}}_{\mathfrak{c}_1}\right] f(x) = \left[\begin{array}{c} 9.433333\times10^{-8}\frac{\Gamma(4)}{\Gamma(4-\mathfrak{c})}x^{3-\mathfrak{c}} + 0.001149\frac{\Gamma(3)}{\Gamma(3-\mathfrak{c})}x^{2-\mathfrak{c}} + 7.256\frac{\Gamma(2)}{\Gamma(2-\mathfrak{c})}x^{1-\mathfrak{c}} \\ -60064.822020\frac{\Gamma(1)}{\Gamma(1-\mathfrak{c})}x^{-\mathfrak{c}} \end{array} \right]. \quad (35)$$

The non-linear model in (34) arises from an energy balance formulation for computing the adiabatic flame temperature, where the variable x represents the equilibrium combustion temperature under idealized adiabatic conditions. The polynomial structure reflects temperature-dependent contributions of heat capacity and enthalpy, leading to inherent non-linearity in the governing equation. The real root corresponds to a physically admissible flame temperature, while the complex roots have no thermodynamic interpretation. Accurate determination of this root is essential for assessing combustion efficiency, thermal stability, and environmental impact, particularly in relation to NO_x formation and system safety. This makes the problem a suitable test case for evaluating memory-based fractional iterative schemes in realistic thermodynamic applications.

It is evident from Tables 3 and 4 that our approach outperforms the state-of-the-art in terms of residual error on both stopping criteria over a range of fractional parameter values $\mathfrak{c}$. As $\mathfrak{c}$ gets closer to 1, the fractional schemes' convergence reaches its maximum.

Table 3. Numerical outcomes of iterative techniques for solving (34) using the criterion $\left|x^{[\kappa+1]} - x^{[\kappa]}\right|$.

Scheme	$\mathfrak{c} = 0.1$	$\mathfrak{c} = 0.3$	$\mathfrak{c} = 0.5$	$\mathfrak{c} = 0.7$	$\mathfrak{c} = 0.9$	CPU-Time
$KFM_1^{[*]}$	2.51×10^{-11}	1.3×10^{-27}	8.3×10^{-33}	2.84×10^{-55}	2.8×10^{-91}	3.4432
$KFM_2^{[*]}$	2.82×10^{-5}	2.8×10^{-19}	8.6×10^{-37}	2.81×10^{-51}	2.8×10^{-79}	2.0768
$KFM_3^{[*]}$	2.83×10^{-13}	3.9×10^{-22}	6.2×10^{-34}	2.83×10^{-49}	2.8×10^{-83}	2.4027
$KFM_4^{[*]}$	2.82×10^{-11}	4.9×10^{-25}	4.8×10^{-46}	2.82×10^{-61}	2.8×10^{-97}	3.9570
$FSM_1^{[**]}$	2.83×10^{-10}	3.39×10^{-29}	5.83×10^{-40}	2.83×10^{-71}	2.83×10^{-101}	1.6538

Table 4. Numerical outcomes of schemes for solving (34) using criteria $\left|f\left(x^{[\kappa]}\right)\right|$.

Scheme	$\mathfrak{c} = 0.1$	$\mathfrak{c} = 0.3$	$\mathfrak{c} = 0.5$	$\mathfrak{c} = 0.7$	$\mathfrak{c} = 0.9$	CPU-Time
$KFM_1^{[*]}$	0.51×10^{-13}	0.23×10^{-29}	2.32×10^{-41}	0.84×10^{-108}	9.24×10^{-276}	3.5076
$KFM_2^{[*]}$	0.82×10^{-10}	2.83×10^{-19}	6.67×10^{-46}	3.01×10^{-101}	1.58×10^{-265}	2.8576
$KFM_3^{[*]}$	8.80×10^{-13}	7.97×10^{-22}	6.02×10^{-47}	5.03×10^{-111}	9.87×10^{-254}	2.4732
$KFM_4^{[*]}$	4.02×10^{-14}	4.09×10^{-25}	4.58×10^{-42}	7.62×10^{-107}	6.28×10^{-219}	2.7657
$FSM_1^{[**]}$	9.03×10^{-17}	1.39×10^{-39}	5.83×10^{-56}	5.83×10^{-117}	0.83×10^{-310}	1.6053

Consistency Analysis: *This section examines the overall effectiveness and convergence of the fractional-order iterative approach for solving the non-linear Equation (1). The investigation includes a thorough examination of convergence behavior, accuracy, and computational efficiency. The proposed fractional-order system converges to the classical ordinary method as the fractional order parameter $\mathfrak{c}$. The scheme's maximum convergence rate and computing efficiency are reached at this key value of $\mathfrak{c} \approx 1$, which keeps the benefits of fractional-order techniques while keeping it similar to conventional methods.*

This criterion not only verifies that the fractional-order scheme operates similarly to the conventional technique at $\mathfrak{c} \approx 1$, but also demonstrates its superior performance in a wider range of applications when $\mathfrak{c}$ is tailored for specific problems. The consistency analysis provides a clear benchmark for comparing the proposed method to existing methods in the literature, demonstrating that it improves stability, accelerates convergence and increases accuracy, especially for problems where traditional methods struggle or fail to converge efficiently. In comparison to conventional techniques, the results clearly illustrate the proposed technique's efficiency and robustness in solving complex non-linear problems. Its practical advantages include increased precision, faster convergence, and consistency in a variety of test situations. Table 5 provides significant numerical evidence to support this claim.

Table 5. Numerical outcomes of schemes for solving (34).

Metrics	$KFM_1^{[*]}$	$KFM_2^{[*]}$	$KFM_3^{[*]}$	$KFM_4^{[*]}$	$FSM_1^{[**]}$
$\left\|x^{[\kappa+1]} - x^{[\kappa]}\right\|$	1.35×10^{-59}	6.13×10^{-56}	5.34×10^{-63}	1.33×10^{-65}	6.3×10^{-74}
$\left\|f(x^{[\kappa]})\right\|$	7.4×10^{-215}	1.76×10^{-235}	2.65×10^{-211}	4.75×10^{-214}	1.42×10^{-319}

The numerical results of the fractional schemes $KFM_1^{[*]}$–$KFM_4^{[*]}$, $FSM_1^{[**]}$ for solving (34) are provided in Tables 5 and 6. Different fractional parameters are considered in the experiments. It is evident from the results that CPU time drops significantly considering error-consecutive and functional values when $\mathfrak{c}$ values rise from 0.1 to 0.9. In addition to this, the residual error, CPU time, and computational order of convergence for various fractional parameter values are shown in Figures 3 and 4a,b. Results in Table 5, indicating that the developed method $FSM_1^{[**]}$ is superior $KFM_1^{[*]}$–$KFM_4^{[*]}$ in terms of errors Figure 3.

Table 6. Overall performance of the fractional schemes for solving (34).

Metrics	$KFM_1^{[*]}$	$KFM_2^{[*]}$	$KFM_3^{[*]}$	$KFM_4^{[*]}$	$FSM_1^{[**]}$
Per-convergence	45.76%	34.76%	49.76%	61.65%	89.67%
Iterations	29	25	27	23	17
CPU-time	4.546	4.675	3.689	4.894	1.790

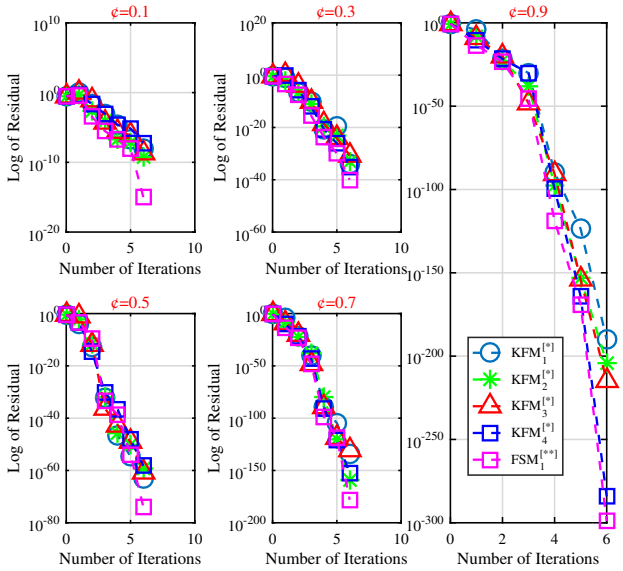


Figure 3. Logarithmic residual error versus iteration count for fractional-order schemes applied to Equation (34) at different fractional orders $\not{c}$. The plots illustrate the effect of $\not{c}$ on convergence rate and numerical efficiency.

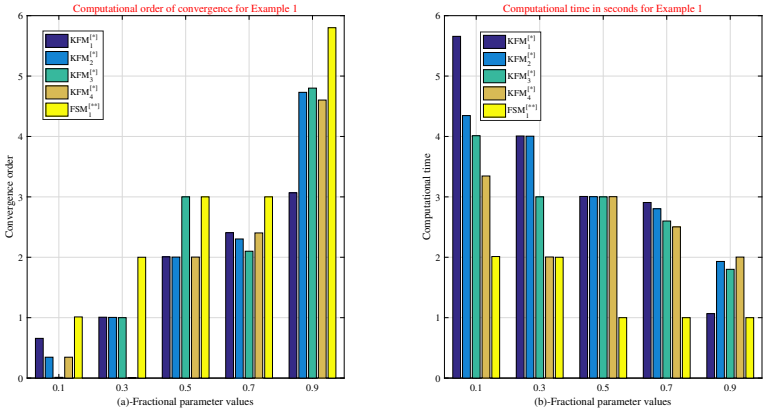


Figure 4. Consistency analysis for Example 1. Panel (**a**) reports the observed computational order of convergence, while panel (**b**) presents the corresponding CPU time (in seconds), for different fractional parameter values $\not{c}$. The comparison highlights the accuracy–efficiency trade-off of the considered fractional methods in solving Equation (34).

Example 2. ***Lennard-Jones Potential in Computational Chemistry***

In computational chemistry, the interaction between two neutral atoms or molecules is described mathematically by the Lennard-Jones potential [37] as

$$f(x) = -12\frac{\vartheta^{12}}{x^{13}} + 6\frac{\vartheta^{5}}{x^{7}}, \tag{36}$$

where x represents the distance between particles and ϑ is the distance at which the potential is zero. The first term represents a short distance repulsion caused by overlapping electron clouds, whereas the second term represents attraction caused by van der Waals forces. The model is widely used in computational chemistry and molecular dynamics to predict how gases, liquids, and solids behave. However, the model has some drawbacks, considering that it oversimplifies real-world atomic

interactions and fails to account for many-body effects. Despite its simplicity, the Lennard-Jones potential is still essential to comprehending intermolecular forces. Taking $\vartheta = 2$ *in (36) it becomes*

$$f(x) = -12\frac{2^{12}}{x^{13}} + 6\frac{2^5}{x^7}. \tag{37}$$

The derivative of Equation (37) of the Caputo type is

$$[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}]f(x) = 6 \times 2^5 \frac{2\Gamma(7)}{\Gamma(7-\mathfrak{c})}x^{6-\mathfrak{c}} - 12 \times 2^{12}\frac{\Gamma(1)}{\Gamma(1-\mathfrak{c})}x^{-\mathfrak{c}}. \tag{38}$$

The Lennard-Jones potential describes the balance between short-range repulsion and long-range attraction, with its stationary point representing the equilibrium interatomic distance. For $\vartheta = 2$*, Equation (37) has a unique positive root at* $x = 2.2449240966187459$*. The initial guess* $x^{[0]} = 1.9$ *is selected through a coarse functional scan near this equilibrium region, ensuring inclusion within the attraction basin of the real root and satisfaction of local convergence conditions.*

The exact root of (37) is 2.2449240966187459 *and the initial guess* $x^{[0]} = 1.9$ *is selected through a coarse functional scan, ensuring proximity to the real root and compliance with local convergence conditions.*

Tables 7 and 8 demonstrate that our method outperforms other methods in terms of residual error on both stopping criteria for different fractional parameter values $\mathfrak{c}$*. When* $\mathfrak{c}$ *gets closer to 1, the fractional schemes' convergence reaches its maximum.*

Table 7. Numerical outcomes of schemes for solving (37) using the criterion $\left|x^{[\kappa+1]} - x^{[\kappa]}\right|$.

Scheme	$\mathfrak{c} = 0.1$	$\mathfrak{c} = 0.3$	$\mathfrak{c} = 0.5$	$\mathfrak{c} = 0.7$	$\mathfrak{c} = 0.9$	CPU-Time
$KFM_1^{[*]}$	2.51×10^{-25}	1.13×10^{-33}	1.3×10^{-44}	8.4×10^{-48}	9.51×10^{-65}	5.5761
$KFM_2^{[*]}$	8.2×10^{-22}	7.83×10^{-36}	8.69×10^{-40}	4.01×10^{-46}	5.02×10^{-71}	5.8762
$KFM_3^{[*]}$	8.83×10^{-23}	6.39×10^{-32}	5.2×10^{-40}	3.2×10^{-52}	3.03×10^{-57}	4.4321
$KFM_4^{[*]}$	9.82×10^{-20}	2.94×10^{-31}	8.1×10^{-42}	6.65×10^{-51}	1.76×10^{-68}	4.6572
$FSM_1^{[**]}$	4.83×10^{-31}	1.35×10^{-43}	7.83×10^{-55}	3.0×10^{-62}	8.45×10^{-97}	3.6533

Table 8. Numerical outcomes of schemes for solving (37) using the criterion $\left|f\left(x^{[\kappa]}\right)\right|$.

Scheme	$\mathfrak{c} = 0.1$	$\mathfrak{c} = 0.3$	$\mathfrak{c} = 0.5$	$\mathfrak{c} = 0.7$	$\mathfrak{c} = 0.9$	CPU-Time
$KFM_1^{[*]}$	5.09×10^{-31}	8.63×10^{-49}	3.7×10^{-91}	7.53×10^{-150}	7.95×10^{-210}	4.5769
$KFM_2^{[*]}$	6.67×10^{-25}	2.88×10^{-51}	9.96×10^{-86}	3.54×10^{-161}	9.87×10^{-211}	5.8766
$KFM_3^{[*]}$	3.13×10^{-34}	5.9×10^{-33}	6.24×10^{-77}	7.47×10^{-141}	8.0×10^{-228}	3.2228
$KFM_4^{[*]}$	6.81×10^{-23}	9.7×10^{-46}	4.58×10^{-76}	3.37×10^{-151}	5.90×10^{-241}	3.6070
$FSM_1^{[**]}$	7.73×10^{-38}	3.39×10^{-59}	5.83×10^{-116}	9.83×10^{-191}	1.13×10^{-291}	2.6039

Consistency Analysis: *The overall stability and effectiveness of the fractional-order iterative approach are examined for solving non-linear Equation (1) in the consistency analysis. The proposed fractional-order system converges to the classical ordinary method as the fractional order parameter* $\mathfrak{c}$ *approaches 1. The scheme reaches its maximum convergence rate and computing efficiency at this key value of* $\mathfrak{c} \approx 1$*, which allows it to preserve the benefits of fractional-order techniques while being comparable to conventional methods.*

In addition to confirming that the fractional-order scheme performs similarly to the traditional technique at $\mathfrak{c} \approx 1$*, this criterion also demonstrates that, when* $\mathfrak{c}$ *is tailored for particular issues, it performs better in a wider range of applications. The consistency analysis establishes a clear benchmark for comparing the proposed method to existing methods in the literature, demonstrating*

better stability, faster convergence, and higher accuracy, especially for problems where traditional methods struggle or fail to converge. The outcomes (see Table 9) highlight the method's potential for outperforming existing techniques in solving complex non-linear equations.

Table 9. Numerical outcomes of schemes for solving (37).

Metrics	$KFM_1^{[*]}$	$KFM_2^{[*]}$	$KFM_3^{[*]}$	$KFM_4^{[*]}$	$FSM_1^{[**]}$
$\left\|x^{[\kappa+1]} - x^{[\kappa]}\right\|$	7.54×10^{-55}	8.38×10^{-59}	3.97×10^{-51}	3.5×10^{-68}	4.32×10^{-96}
$\left\|f\left(x^{[\kappa]}\right)\right\|$	1.65×10^{-208}	9.87×10^{-211}	8.0×10^{-238}	5.90×10^{-261}	1.13×10^{-321}

The numerical results of the fractional schemes $KFM_1^{[*]}$–$KFM_4^{[*]}$, $FSM_1^{[**]}$ for solving (37) are provided in Tables 9 and 10. Different fractional parameters are considered in the experiments. It is evident from the results that CPU time drops significantly considering error-consecutive and functional values when ϕ values rise from 0.1 to 0.9. In addition to this, the residual error, CPU time, and computational order of convergence for various fractional parameter values are shown in Figures 5 and 6a,b. Results in Table 9, indicating that the developed method $FSM_1^{[**]}$ is superior $KFM_1^{[*]}$–$KFM_4^{[*]}$ in terms of errors Figure 5.

Table 10. Overall performance of the fractional schemes for solving (37).

Metrics	$KFM_1^{[*]}$	$KFM_2^{[*]}$	$KFM_3^{[*]}$	$KFM_4^{[*]}$	$FSM_1^{[**]}$
Per-convergence	43.12%	41.89%	51.06%	57.96%	87.03%
Iterations	27	23	26	24	19
CPU-time	5.654	5.132	3.998	3.107	1.556

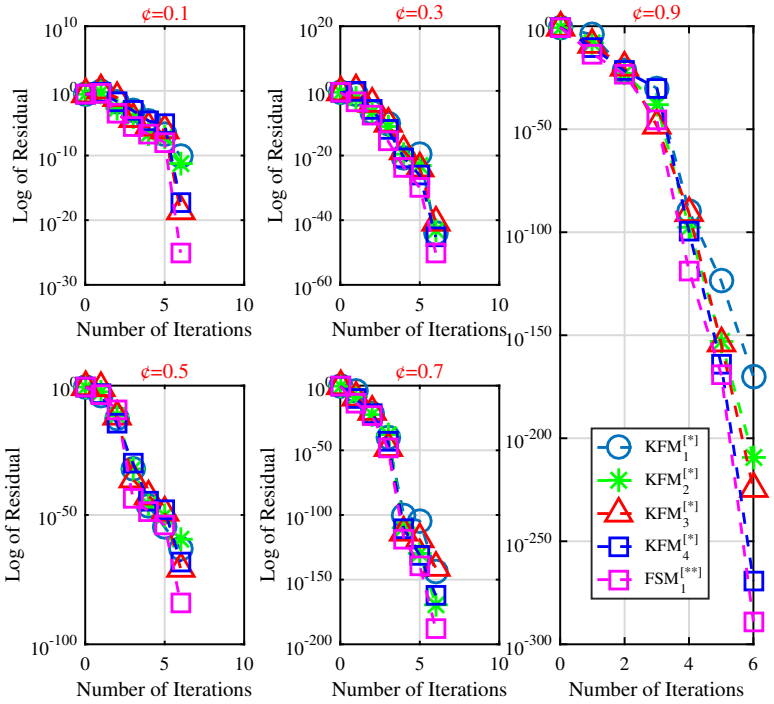


Figure 5. Logarithmic residual error versus iteration count for fractional-order schemes applied to Equation (37) at different fractional orders ϕ. The plots illustrate the effect of ϕ on convergence rate and numerical efficiency.

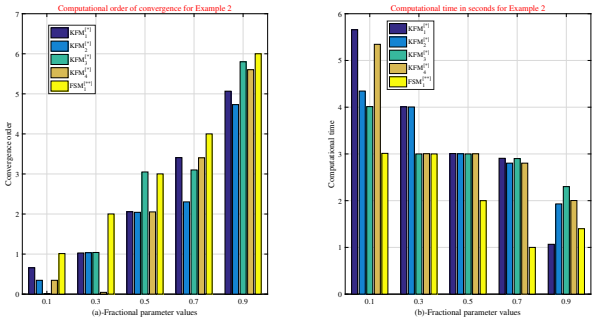


Figure 6. Consistency analysis for Example 2. Panel (**a**) reports the observed computational order of convergence, while panel (**b**) presents the corresponding CPU time (in seconds), for different fractional parameter values $\mathfrak{c}$. The comparison highlights the accuracy–efficiency trade-off of the considered fractional methods in solving Equation (37).

Example 3. ***Fractional-Order Differential Equations in Engineering Control Systems***

Caputo-type fractional differential equations are essential for simulating memory and hereditary-effect systems, which take into account the dynamics that cannot be captured by conventional integer-order models. This is often the case in applications like viscoelasticity, anomalous diffusion, and control systems. Offering greater flexibility by accounting for non-local interactions makes this method suitable for explaining complex phenomena and processes. Notably, the Caputo derivative allows for more physically realistic initial circumstances. These equations are essential for modeling phenomena and processes where previous states influence the present behavior. Overall, more precise models are required for intricate systems with the following non-linear equations:

$$\begin{cases} \left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] h(x) + e^{h(x)} = 0, \\ h(0) = 0, h'(0) = 0, 1 < \mathfrak{c} \leq 2, x \geq 0 \end{cases} \tag{39}$$

Equation (39) is simulated using the following polynomial as suggested by [38]:

$$h_n(x) \approx \left[-\frac{1}{\Gamma(\mathfrak{c}+1)} x^{\mathfrak{c}} + \frac{1}{\Gamma(3\mathfrak{c}+1)} x^{2\mathfrak{c}} + \left(\frac{2(\Gamma(\mathfrak{c}+1))^2 \Gamma(2\mathfrak{c}+1)}{2\Gamma^2(\mathfrak{c}+1)\Gamma(3\mathfrak{c}+1)} \right) x^{3\mathfrak{c}} + \ldots \right]. \tag{40}$$

For $\mathfrak{c} = 1$, Equation (41) simplifies to

$$f(x) \approx -x + \frac{1}{2}x^2 - 2x^3, \tag{41}$$

Equation (41) has the following solutions:

$$\zeta_1 = 0.0, \zeta_{2,3} = 0.125 \pm 0.695i. \tag{42}$$

The initial guess $x^{[0]} = 0.08$ is selected through a coarse functional scan, ensuring proximity to the real root and compliance with local convergence conditions. The derivative of Equation (41) of the Caputo type is

$$\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(x) = \left[-\frac{2\Gamma(4)}{\Gamma(4-\mathfrak{c})} x^{3-\mathfrak{c}} + \frac{1}{2}\frac{\Gamma(3)}{\Gamma(3-\mathfrak{c})} x^{2-\mathfrak{c}} - \frac{\Gamma(2)}{\Gamma(2-\mathfrak{c})} x^{1-\mathfrak{c}} \right]. \tag{43}$$

Tables 11 and 12 demonstrate that our method outperforms other methods in terms of residual error on both stopping criteria for different fractional parameter values $\mathfrak{c}$. When $\mathfrak{c}$ gets closer to 1, the fractional schemes' convergence reaches its maximum.

Table 11. Numerical outcomes of schemes for solving (41) using criterion $\left|x^{[\kappa+1]} - x^{[\kappa]}\right|$.

Scheme	$\mathfrak{c} = 0.1$	$\mathfrak{c} = 0.3$	$\mathfrak{c} = 0.5$	$\mathfrak{c} = 0.7$	$\mathfrak{c} = 0.9$	CPU-Time
$KFM_1^{[*]}$	5.10×10^{-14}	1.43×10^{-29}	5.53×10^{-40}	2.84×10^{-51}	3.24×10^{-121}	5.576
$KFM_2^{[*]}$	1.82×10^{-15}	2.84×10^{-21}	8.46×10^{-46}	5.81×10^{-51}	2.63×10^{-131}	5.876
$KFM_3^{[*]}$	1.30×10^{-12}	3.39×10^{-22}	6.24×10^{-47}	6.83×10^{-64}	5.30×10^{-122}	4.432
$KFM_4^{[*]}$	1.20×10^{-11}	9.50×10^{-26}	5.80×10^{-46}	4.82×10^{-61}	9.43×10^{-141}	4.657
$FSM_1^{[**]}$	2.03×10^{-18}	1.39×10^{-39}	1.83×10^{-56}	7.83×10^{-81}	7.83×10^{-151}	3.653

Table 12. Numerical outcomes of schemes for solving (41) using the criterion $\left|f\left(x^{[\kappa]}\right)\right|$.

Scheme	$\mathfrak{c} = 0.1$	$\mathfrak{c} = 0.3$	$\mathfrak{c} = 0.5$	$\mathfrak{c} = 0.7$	$\mathfrak{c} = 0.9$	CPU-Time
$KFM_1^{[*]}$	2.51×10^{-14}	1.03×10^{-39}	8.23×10^{-70}	0.84×10^{-134}	7.78×10^{-231}	3.4570
$KFM_2^{[*]}$	7.52×10^{-17}	2.80×10^{-51}	8.36×10^{-76}	2.01×10^{-154}	2.68×10^{-232}	3.8076
$KFM_3^{[*]}$	6.63×10^{-16}	3.90×10^{-42}	6.42×10^{-77}	0.83×10^{-143}	6.80×10^{-243}	3.4002
$KFM_4^{[*]}$	2.52×10^{-17}	4.09×10^{-45}	4.28×10^{-76}	9.82×10^{-141}	8.89×10^{-254}	3.3607
$FSM_1^{[**]}$	0.03×10^{-19}	3.39×10^{-59}	5.53×10^{-96}	2.03×10^{-191}	7.83×10^{-298}	2.3603

Consistency Analysis: *Our consistency research looks into the overall performance and stability of the fractional-order iterative approach to solving non-linear Equation (1). The proposed fractional-order system converges to the classical ordinary method as the fractional order parameter* $\mathfrak{c}$*. The scheme's maximum convergence rate and computing efficiency are reached at this key value of* $\mathfrak{c} \approx 1$*, which keeps the benefits of fractional-order techniques while keeping it similar to conventional methods. According to this criterion, the fractional-order scheme performs better in a wider range of applications when* $\mathfrak{c}$ *is tuned for particular problems, in addition to confirming that it operates similarly to the traditional technique at* $\mathfrak{c} \approx 1$*. The consistency analysis establishes a clear benchmark for comparing the proposed method to existing methods in the literature, demonstrating that it provides better stability, faster convergence, and higher accuracy, particularly for problems where traditional methods struggle or fail to converge. The outcomes (see Table 13) highlight how reliable the method is and how it can be used to solve complicated non-linear equations more efficiently than existing techniques.*

Table 13. Numerical outcomes of schemes for solving (41).

Metrics	$KFM_1^{[*]}$	$KFM_2^{[*]}$	$KFM_3^{[*]}$	$KFM_4^{[*]}$	$FSM_1^{[**]}$
$\left\|x^{[\kappa+1]} - x^{[\kappa]}\right\|$	0.32×10^{-86}	1.36×10^{-99}	0.34×10^{-91}	2.35×10^{-96}	1.36×10^{-126}
$\left\|f\left(x^{[\kappa]}\right)\right\|$	7.54×10^{-211}	0.12×10^{-208}	9.35×10^{-292}	2.33×10^{-287}	9.32×10^{-308}

Considering the fractional schemes' $FSM_1^{[**]}$, $KFM_1^{[*]}$–$KFM_4^{[*]}$ *the numerical results for solving (41) for different fractional parameters are displayed in Tables 13 and 14. The table demonstrates that CPU time is reduced considering error-consecutive and functional values when fractional parameter values are increased from 0.1 to 0.9 as present in Figure 7. Figure 8a displays the computing order of convergence, residual error Figure 7, and CPU time Figure 8b to evaluate this for other fractional parameter values. Additionally, Tables 13 and 14 is provided to show that the developed method,* $FSM_1^{[**]}$*, outperforms the existing methods,* $KFM_1^{[*]}$–$KFM_4^{[*]}$*, in terms of errors.*

Table 14. Overall performance of the fractional schemes for solving (41).

Metrics	$\mathbf{KFM}_1^{[*]}$	$\mathbf{KFM}_2^{[*]}$	$\mathbf{KFM}_3^{[*]}$	$\mathbf{KFM}_4^{[*]}$	$\mathbf{FSM}_1^{[**]}$
Per-convergence	55.03%	45.13%	52.53%	57.98%	94.67%
Iterations	31	21	26	27	21
CPU-time	5.576	5.005	4.119	4.654	2.765

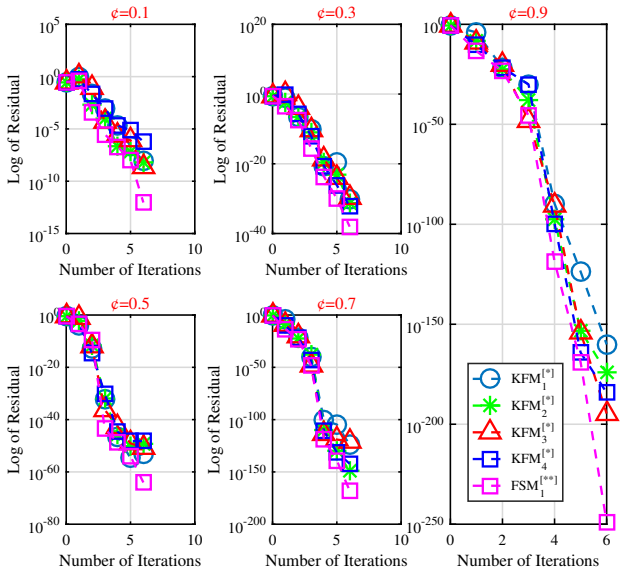


Figure 7. Logarithmic residual error versus iteration count for fractional-order schemes applied to Equation (41) at different fractional orders $\mathscr{c}$. The plots illustrate the effect of $\mathscr{c}$ on convergence rate and numerical efficiency.

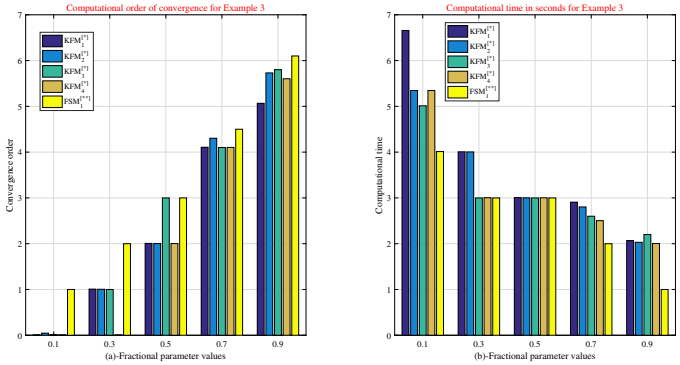


Figure 8. Consistency analysis for Example 3. Panel (**a**) reports the observed computational order of convergence, while panel (**b**) presents the corresponding CPU time (in seconds), for different fractional parameter values $\mathscr{c}$. The comparison highlights the accuracy–efficiency trade-off of the considered fractional methods in solving Equation (41).

Example 4. ***Non-linear Demand Inventory Management Model in Operations Research***

The inventory management model [39] with non-linear demand is essential for optimizing inventories in rapidly changing and price-sensitive markets. This is especially true for managing inventory in E-Commerce [40]. The model accounts for the complex demand-price relationship while offering a realistic simulation of market behaviors, which assists in better demand forecasting and resource allocation. It also aids in balancing inventory holding, ordering, and penalty costs, resulting in more efficient inventory control. Overall, this strategy promotes sustainable company operations by decreasing waste and increasing customer satisfaction through more effective stock management.

The demand $D^{[*]}(p)$, which is modeled as a non-linear function of the time and the good's price is examined in an inventory system. The cost function consists of several factors, including

- *Inventory holding costs ($C_h^{[*]}$)*
- *Order costs per restocking inventory ($C_0^{[*]}$)*
- *Cost of penalties for back-orders ($C_p^{[*]}$)*

The objective of this model is to determine the optimal reorder quantity x such that price p is given as

$$D^{[*]}(p) = a_3p^3 + a_2p^2 + a_1p + a_0p^0, \tag{44}$$

where a_3, a_2, a_1, and a_0 are market structure-derived constants. The cost function includes non-linear terms, given that dependency on reorder is x and the definition of the good is given.

$$TC^{[*]}(x) = C_h^{[*]}\left(\frac{x}{2.0}\right) + C_0^{[*]} \times \frac{D^{[*]}(x)}{x} + C_p^{[*]} \times B(x), \tag{45}$$

where $B(x)$ is the back order function and modeled as:

$$B(x) = \frac{1}{2}\left(x - D^{[*]}(p)\right)^2. \tag{46}$$

To find the optimal reorder quantity x, the total cost function is differentiated considering x

$$\frac{d}{dx}\left(C_h^{[*]}\left(\frac{x}{2}\right) + C_0^{[*]}\frac{D^{[*]}(P)}{Q} + C_p^{[*]}B(x)\right) = 0. \tag{47}$$

This implies

$$\frac{C_h^{[*]}}{2} - C_0^{[*]}\left(\frac{\left(\ a_3p^3 + a_2p^2 + a_1p + a_0p^0\ \right)}{x^2}\right) + C_p^{[*]}\left(x - \left(\ a_3p^3 + a_2p^2 + a_1p + a_0p^0\ \right)\right) = 0. \tag{48}$$

After rearranging and collecting the term contain x^3, x^2, and constant, we have

$$C_p^{[*]}x^3 + \left(\frac{C_h^{[*]}}{2} - C_p^{[*]}\left(\ a_3p^3 + a_2p^2 + a_1p + a_0p^0\ \right)\right)x^2 - C_0^{[*]}\left(\ a_3p^3 + a_2p^2 + a_1p + a_0p^0\ \right) = 0. \tag{49}$$

Taking $C_p^{[*]} = 5, C_h^{[*]} = 1.5, C_0^{[*]} = 250$, a reasonable penalty cost is considered to deter stock shortages without being excessively severe. For a price-sensitive market where demand declines non-linearly as price increases, we select $a_3 = -0.2$, $a_2 = 2.5, a_1 = -3.0, a_0 = 150, p = 10$.

$$f(x) \approx 5x^3 - 849.25x^2 - 42500 = 0. \tag{50}$$

From an operational perspective, the non-linear Equation (50) represents the equilibrium condition between inventory holding, ordering, and back-order penalty costs under a price-sensitive demand structure. Although the resulting model reduces to a scalar non-linear equation, the incorporation of fractional iterative schemes is motivated by the inherent memory effects present in real inventory systems, such as delayed demand response, accumulated market trends, and historical pricing influence. The fractional order introduces a controlled memory mechanism that stabilizes the numerical trajectory toward the optimal reorder quantity, mitigating abrupt oscillations caused by non-linear demand elasticity. Consequently, fractional methods provide not only improved global convergence properties but also a more realistic representation of decision dynamics in inventory management, where past states influence present restocking strategies.

Equation (50) has the following solutions:

$$\zeta_1 = 170.143621316548, \zeta_{2,3} = -0.1468 \pm 7.0665i. \quad (51)$$

The initial guess $x^{[0]} = 170.50$ *is selected through a coarse functional scan, ensuring proximity to the real root and compliance with local convergence conditions. The derivative of Equation (50) of the Caputo type is*

$$\left[\eth_{\mathfrak{c}_1}^{\mathfrak{c}}\right] f(x) = \left[\frac{5\Gamma(4)}{\Gamma(4-\mathfrak{c})} x^{3-\mathfrak{c}} - \frac{849.25\Gamma(3)}{\Gamma(3-\mathfrak{c})} x^{2-\mathfrak{c}} - \frac{42500\Gamma(1)}{\Gamma(1-\mathfrak{c})} x^{-\mathfrak{c}} \right]. \quad (52)$$

Tables 15 and 16 demonstrates that our solution outperforms those by previous methods considering residual error on both stopping criteria for different fractional parameter values $\mathfrak{c}$*. The convergence of fractional schemes peaks as* $\mathfrak{c}$ *approaches 1.*

Table 15. Numerical outcomes of schemes for solving (50) using criteria $\left|x^{[\kappa+1]} - x^{[\kappa]}\right|$.

Scheme	$\mathfrak{c}$ = 0.1	$\mathfrak{c}$ = 0.3	$\mathfrak{c}$ = 0.5	$\mathfrak{c}$ = 0.7	$\mathfrak{c}$ = 0.9	CPU-Time
$KFM_1^{[*]}$	2.61×10^{-5}	1.66×10^{-13}	8.03×10^{-30}	5.74×10^{-41}	8.89×10^{-57}	5.5436
$KFM_2^{[*]}$	3.62×10^{-7}	2.38×10^{-11}	0.60×10^{-36}	5.51×10^{-46}	7.11×10^{-61}	3.8736
$KFM_3^{[*]}$	1.83×10^{-6}	3.93×10^{-12}	6.20×10^{-37}	0.09×10^{-45}	6.82×10^{-73}	4.0042
$KFM_4^{[*]}$	5.12×10^{-5}	4.59×10^{-15}	4.98×10^{-36}	5.02×10^{-41}	5.43×10^{-68}	9.0007
$FSM_1^{[**]}$	1.11×10^{-11}	3.09×10^{-39}	0.83×10^{-56}	4.83×10^{-61}	2.53×10^{-91}	2.6634

Table 16. Numerical outcomes of schemes for solving (50) using criteria $\left|f\left(x^{[\kappa]}\right)\right|$.

Scheme	$\mathfrak{c}$ = 0.1	$\mathfrak{c}$ = 0.3	$\mathfrak{c}$ = 0.5	$\mathfrak{c}$ = 0.7	$\mathfrak{c}$ = 0.9	CPU-Time
$KFM_1^{[*]}$	9.51×10^{-14}	1.3×10^{-29}	8.3×10^{-30}	0.22×10^{-137}	1.18×10^{-189}	5.576
$KFM_2^{[*]}$	8.82×10^{-11}	2.8×10^{-21}	8.6×10^{-36}	8.31×10^{-141}	9.8×10^{-215}	5.876
$KFM_3^{[*]}$	0.83×10^{-13}	3.9×10^{-22}	6.2×10^{-37}	8.83×10^{-131}	0.89×10^{-241}	4.432
$KFM_4^{[*]}$	0.02×10^{-12}	4.9×10^{-25}	4.8×10^{-36}	5.82×10^{-113}	9.86×10^{-208}	4.657
$FSM_1^{[**]}$	4.83×10^{-17}	3.39×10^{-39}	5.83×10^{-56}	5.83×10^{-158}	0.80×10^{-321}	3.653

Consistency analysis: *In consistency analysis, the overall stability of the fractional-order iterative methods is discussed for solving (50) using* $\mathfrak{c} \approx 1$*. At this value, the fractional order scheme becomes the ordinary and has a maximum convergence rate and efficiency. This criterion also indicates how much the proposed method is better than existing methods in the literature.*

This criterion not only verifies that the fractional-order scheme operates similarly to the conventional technique at $\mathfrak{c} \approx 1$*, but also demonstrates its superior performance in a wider range of applications when* $\mathfrak{c}$ *is tailored for specific problems. The consistency analysis provides a clear benchmark for comparing the proposed method with the existing methods in the literature, demonstrating that it improves stability, accelerates convergence, and increases accuracy, especially for problems where traditional methods struggle or fail to converge efficiently. In comparison to existing methods* $KFM_1^{[*]}$–$KFM_4^{[*]}$*, the results demonstrate the developed method's* $FSM_1^{[**]}$ *robustness for solving complex non-linear equations more efficiently (see Table 17).*

The numerical results of the fractional schemes $KFM_1^{[*]}$–$KFM_4^{[*]}$, $FSM_1^{[**]}$ for solving (34) are provided in Tables 17 and 18. Different fractional parameters are considered in the experiments. It is evident from the results that CPU time drops significantly considering error-consecutive and functional values when $\mathfrak{c}$ values rise from 0.1 to 0.9, as illustrated in Figure 9. In addition to this, the residual error, CPU time, and computational order of

convergence for various fractional parameter values are shown in Figure 10a,b. Results in Table 17, indicating that the developed method $FSM_1^{[**]}$ is superior $KFM_1^{[*]}$–$KFM_4^{[*]}$ in terms of errors Figure 9.

Table 17. Numerical outcomes of schemes for solving (50).

Metrics	$KFM_1^{[*]}$	$KFM_2^{[*]}$	$KFM_3^{[*]}$	$KFM_4^{[*]}$	$FSM_1^{[**]}$
$\left\|x^{[\kappa+1]} - x^{[\kappa]}\right\|$	7.98×10^{-67}	1.76×10^{-58}	0.90×10^{-53}	8.98×10^{-61}	0.09×10^{-97}
$\left\|f\left(x^{[\kappa]}\right)\right\|$	5.32×10^{-215}	8.98×10^{-213}	6.55×10^{-217}	9.09×10^{-219}	7.72×10^{-320}

Table 18. Overall performance of the fractional schemes for solving (34).

Metrics	$KFM_1^{[*]}$	$KFM_2^{[*]}$	$KFM_3^{[*]}$	$KFM_4^{[*]}$	$FSM_1^{[**]}$
Per-convergence	55.89%	54.23%	52.78%	45.05%	91.90%
Iterations	23	29	21	20	15
CPU-time	3.675	6.098	5.876	2.554	0.986

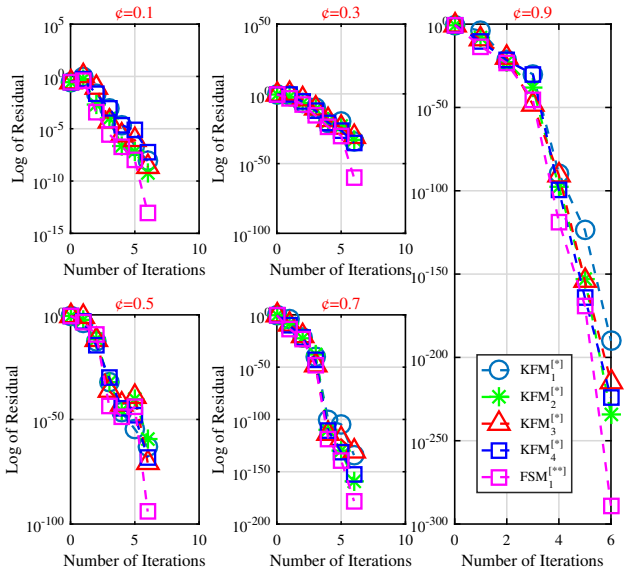


Figure 9. Logarithmic residual error versus iteration count for fractional-order schemes applied to Equation (50) at different fractional orders $\not{c}$. The plots illustrate the effect of $\not{c}$ on convergence rate and numerical efficiency.

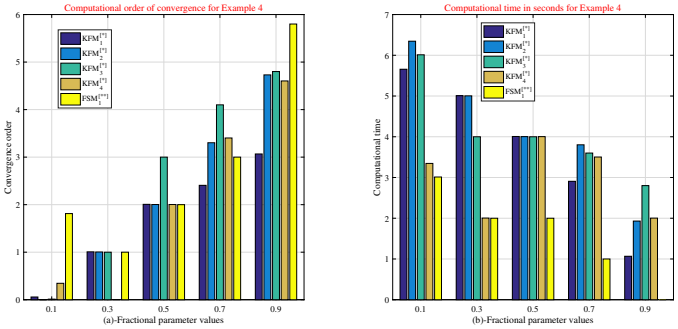


Figure 10. Consistency analysis for Example 4. Panel (**a**) reports the observed computational order of convergence, while panel (**b**) presents the corresponding CPU time (in seconds), for different fractional parameter values $\not{c}$. The comparison highlights the accuracy–efficiency trade-off of the considered fractional methods in solving Equation (50).

Physical interpretation and practical relevance. Although the inventory optimization problem ultimately reduces to a scalar non-linear equation, the use of a fractional-order iterative framework is motivated by the intrinsic memory effects observed in real inventory systems, such as delayed demand adjustment, cumulative pricing influence, and historical stock imbalance. The fractional parameter introduces a controlled memory mechanism that enhances numerical stability and reflects realistic decision dynamics beyond classical memoryless solvers.

- **Memory-driven demand adjustment:** The fractional formulation captures delayed market responses, where current reorder decisions depend not only on instantaneous demand but also on past pricing and consumption trends.
- **Stabilization of reorder dynamics:** Fractional memory smooths abrupt changes caused by non-linear demand elasticity, leading to a stable convergence toward the optimal reorder quantity and preventing oscillatory inventory policies.
- **Robust cost equilibrium behavior:** The real root ζ_1 represents a physically meaningful equilibrium where holding, ordering, and back-order costs are balanced, while complex roots correspond to economically infeasible inventory states.

4.2. Computational Complexity and Scalability

The efficiency of the suggested method is demonstrated by CPU time comparisons; however, its computational complexity should also be addressed. Using the Caputo fractional derivative is one way where higher-order fractional scheme adds to the computational burden. However, this is mitigated by the advantages listed below:

- Improved convergence: The method achieves a convergence order of $5\mathfrak{c}+1$ while requiring fewer iterations than existing schemes.
- Memory-Efficient Fractional Operators: These operators limit the expansion of memory by performing approximations to the Caputo derivative using efficient quadrature or convolution techniques.
- Scalability: The technique's iterative structure makes it suitable for more complex and larger problems in engineering.
- Dimensional Extension: The scheme can be accelerated to higher-dimensional problems with acceptable complexity levels by utilizing sparse grid or domain decomposition techniques in conjunction with a recently developed technique.

5. Conclusions

Using the Caputo-fractional derivative, this study developed an efficient fractional Caputo-type technique $\mathrm{FSM}_1^{[**]}$, with a convergence order of $5\mathfrak{c}+1$. The concept of basins of attraction was employed to show that the region of convergence showcases better convergence behavior than the existing approaches. Symmetries in the basins indicated the number of iterations. Efficiency and stability were evaluated by considering various non-linear problems in operations research and engineering optimization. The developed method $\mathrm{FSM}_1^{[**]}$ showed superiority to the current methods $\mathrm{KFM}_1^{[*]}$–$\mathrm{KFM}_4^{[*]}$ considering error, CPU time, consistency, and efficiency measures. The strengths of the proposed method are:

- Higher Accuracy: The approach has a higher-order convergence rate, which allows for more accuracy in approximating solutions for non-linear problems, particularly in complex, memory-based systems (see Tables 1–18).
- Enhanced Efficiency: The numerical findings showed that the method substantially reduces residual error Figures 3, 5, 7 and 9, and the computational order of conver-

gence and processing time Figures 4, 6, 8 and 10 when compared to classical methods; this makes the developed method more viable for real-world applications.

- Robustness in Complex Domains: Traditional techniques fail to address problems with fractional derivatives, while our method performed better in applications like anomalous diffusion and viscoelasticity.
- Overall performance: The overall performance of the fractional schemes in Tables 6, 10, 14 and 18 showed that the proposed technique $FSM_1^{[**]}$ outperform existing schemes $KFM_1^{[*]}$–$KFM_4^{[*]}$ in terms of CPU time, percentage convergence, and iteration count, indicating consistency and efficiency.
- Wide Applications: The method can be employed in a variety of domains, from engineering control systems to operations research, making it widely applicable for solving non-linear optimization problems.

Although the proposed higher-order fractional numerical method outperforms the state of the art, some limitations should be noted. The first limitation comes from the scheme's reliance on optimal parameter adjustments (e.g., fractional order and step size), where calculating these factors may be challenging in certain application areas. Second, although higher-order techniques are developed to solve problems in greater dimensions or systems of equations, the computational complexity may increase exponentially. Future research may overcome these challenges by exploring a broader range of non-linear equations and systems, especially those in high-dimensional, real-world applications. The limitations of the proposed method are summarized below.

- Sensitivity to Parameters: Choosing appropriate fractional orders and step sizes is essential to the method's success. This can be challenging in practice.
- Adaptability: The method's ability to handle large-scale or high-dimensional issues may be impacted by its increased computing complexity.

Future research should focus on adaptive parameter selection methodologies, such as data-driven and machine learning, to improve the method's robustness and utility. A deeper theoretical understanding may also be attained through examining the scheme's stability and convergence characteristics under various circumstances. Finally, incorporating parallel computing may reduce the computational time of the developed method and facilitate its industrial applications for real-time decision support.

Author Contributions: M.S. devised the project and developed the main conceptual ideas. M.S. and N.K. formulated the methodology. M.S. developed the software used in the study. M.S. and P.P., performed the validation. M.S., P.P. and N.K. conducted the formal analysis. M.S. and P.P. carried out the investigation and managed resources and data curation. M.S. and N.K. prepared the original draft. P.P. and M.S. reviewed and edited the manuscript. N.K., P.P. and M.S. handled the visualization. N.K. and M.S. supervised the project. P.P. and N.K. managed project administration and secured funding. All authors have read and agreed to the published version of the manuscript.

Funding: Open access funding provided by UiT The Arctic University of Norway.

Data Availability Statement: The data used to support the findings of this study are included within the article.

Acknowledgments: This study did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflicts of Interest: The authors declare that they have no competing financial interests and personal relationships that could have appeared to influence the research in this paper.

Appendix A

Theorem A1 (Generalized fractional Taylor expansion). *Let $0 < \mathscr{c} \leq 1$ and let $f : [\xi_1, \xi_2] \to \mathbb{R}$ be a function such that $f \in C^{(n+1)\mathscr{c}}([\xi_1, \xi_2])$, i.e., the Caputo fractional derivatives $\left[\eth_{\xi_1}^{\gamma\mathscr{c}}\right] f(x)$ exist and are continuous on $[\xi_1, \xi_2]$ for $\gamma = 1, \ldots, n+1$. Here, $[\eth_a^{\mathscr{c}}]$ denotes the Caputo-type fractional derivative of order $\mathscr{c}$ with lower limit a.*

Under these assumptions, the generalized fractional Taylor formula [29] holds and is given by

$$f(x) = \sum_{i=0}^{n} \left[\eth_{\xi_1}^{i\mathscr{c}}\right] f(\xi_1) \frac{(x-\xi_1)^{i\mathscr{c}}}{\Gamma(i\mathscr{c}+1)} + \left[\eth_{\xi_1}^{(n+1)\mathscr{c}}\right] f(\xi) \frac{(x-\xi_1)^{(n+1)\mathscr{c}}}{\Gamma((n+1)\mathscr{c}+1)}, \tag{A1}$$

where

$$\xi_1 \leq \xi \leq x, \quad \forall x \in (\xi_1, \xi_2],$$

and the operator $\left[\eth_{\xi_1}^{n\mathscr{c}}\right]$ is defined by

$$\left[\eth_{\xi_1}^{n\mathscr{c}}\right] = \underbrace{\left[\eth_{\xi_1}^{\mathscr{c}}\right] \circ \left[\eth_{\xi_1}^{\mathscr{c}}\right] \circ \cdots \circ \left[\eth_{\xi_1}^{\mathscr{c}}\right]}_{n \text{ times}}.$$

Using the fractional Taylor expansion about the point ξ, we obtain the following local representation:

$$f(x) = \frac{{}^{C}D_{\xi}^{\alpha} f(\xi)}{\Gamma(\alpha+1)}(x-\xi)^{\alpha} + \frac{{}^{C}D_{\xi}^{2\alpha} f(\xi)}{\Gamma(2\alpha+1)}(x-\xi)^{2\alpha} + O\Big((x-\xi)^{3\alpha}\Big), \tag{A2}$$

where the big-O term denotes a function bounded by $C|x-\xi|^{3\mathscr{c}}$ as $x \to \xi$, for some constant $C > 0$. Factoring out the leading term yields

$$f(x) = \frac{\left[\eth_{\xi}^{\mathscr{c}}\right] f(\xi)}{\Gamma(\mathscr{c}+1)} \Big[(x-\xi)^{\mathscr{c}} + b_2(x-\xi)^{2\mathscr{c}} + O\Big((x-\xi)^{3\mathscr{c}}\Big)\Big], \tag{A3}$$

where

$$b_j = \frac{\Gamma(\mathscr{c}+1)}{\Gamma(j\mathscr{c}+1)} \frac{\left[\eth_{\xi}^{j\mathscr{c}}\right] f(\xi)}{\left[\eth_{\xi}^{\mathscr{c}}\right] f(\xi)}, \quad j = 2, 3, \ldots$$

Consequently, the Caputo-type fractional derivative of $f(x)$ in a neighborhood of ξ admits the expansion

$$\left[\eth_{\xi}^{\mathscr{c}}\right] f(x) = \frac{\left[\eth_{\xi}^{\mathscr{c}}\right] f(\xi)}{\Gamma(\mathscr{c}+1)} \left[\Gamma(\mathscr{c}+1) + \frac{\Gamma(2\mathscr{c}+1)}{\Gamma(\mathscr{c}+1)} b_2 (x-\xi)^{\mathscr{c}}\right] + O\Big((x-\xi)^{2\mathscr{c}}\Big). \tag{A4}$$

These fractional expansions are valid under the stated regularity assumptions and provide the analytical foundation for investigating the convergence, stability, and reliability of the proposed iterative methods.

Appendix B

The coefficients of $e^{[\kappa]}$ were utilized in the proof of Theorem 1.

$$g_1^{[*]} = \left(-\frac{(2^{\mathscr{c}})^4 {b_2}^3 b_3}{(\Gamma(\mathscr{c}))^2 \mathscr{c}^2 \pi} \left(\Gamma\left(\mathscr{c}+\frac{1}{2}\right)\right)^2 + 6\,\frac{{b_2}^5 (2^{\mathscr{c}})^8 (\Gamma(\mathscr{c}+1/2))^4}{\mathscr{c}^4 \pi^2 (\Gamma(\mathscr{c}))^4} + 6\,\frac{{b_2}^5 (2^{\mathscr{c}})^4 (\Gamma(\mathscr{c}+1/2))^2}{(\Gamma(\mathscr{c}))^2 \mathscr{c}^2 \pi} \right),$$

$$g_2^{[*]} = \left(-\frac{(2^{\mathscr{c}})^4 {b_2}^3 b_3}{(\Gamma(\mathscr{c}))^2 \mathscr{c}^2 \pi} \left(\Gamma\left(\mathscr{c} + \frac{1}{2} \right) \right)^2 + 6 \frac{{b_2}^5 (2^{\mathscr{c}})^8 (\Gamma(\mathscr{c}+1/2))^4}{\mathscr{c}^4 \pi^2 (\Gamma(\mathscr{c}))^4} + 6 \frac{{b_2}^5 (2^{\mathscr{c}})^4 (\Gamma(\mathscr{c}+1/2))^2}{(\Gamma(\mathscr{c}))^2 \mathscr{c}^2 \pi} \right),$$

$$g_3^{[*]} = \left(-12 \frac{{b_2}^5 (2^{\mathscr{c}})^6 (\Gamma(\mathscr{c}+1/2))^3}{(\Gamma(\mathscr{c}))^3 \mathscr{c}^3 \pi^{3/2}} + 3 \frac{b_3 {b_2}^3 (2^{\mathscr{c}})^6 (\Gamma(\mathscr{c}+1/2))^3}{(\Gamma(\mathscr{c}))^3 \mathscr{c}^3 \pi^{3/2}} \right),$$

$$g_4^{[*]} = \begin{pmatrix} -6 \frac{b_3 (2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+1/2) {b_2}^3 (3^{\mathscr{c}})^3 \sqrt{3} \Gamma(\mathscr{c}+1/3) \Gamma(\mathscr{c}+2/3)}{(\Gamma(\mathscr{c}))^3 \mathscr{c}^3 \pi^{3/2}} \\ +8 \frac{b_3 {b_2}^3 (3^{\mathscr{c}})^3 \sqrt{3} \Gamma(\mathscr{c}+1/3) \Gamma(\mathscr{c}+2/3)}{(\Gamma(\mathscr{c}))^2 \mathscr{c}^2 \pi} \end{pmatrix},$$

$$g_5^{[*]} = \left(-2 \frac{b_3 {b_2}^3 (3^{\mathscr{c}})^3 \sqrt{3} \Gamma(\mathscr{c}+1/3) \Gamma(\mathscr{c}+2/3)}{\Gamma(\mathscr{c}) \mathscr{c} \sqrt{\pi} (2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+1/2)} - 2 \frac{{b_2} {b_3}^2 (3^{\mathscr{c}})^3 \sqrt{3} \Gamma(\mathscr{c}+1/3) \Gamma(\mathscr{c}+2/3)}{\Gamma(\mathscr{c}) \mathscr{c} \sqrt{\pi} (2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+1/2)} \right),$$

$$g_6^{[*]} = \left(-4 \frac{(2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+1/2) {b_2}^2 b_4}{\Gamma(\mathscr{c}) \mathscr{c} \sqrt{\pi}} + 4 {b_2}^2 b_4 + 2 {b_2} {b_3}^2 \right),$$

$$g_7^{[*]} = \begin{pmatrix} +\frac{b_3 (2^{\mathscr{c}})^2 {b_2}^3}{\Gamma(\mathscr{c}) \mathscr{c} \sqrt{\pi}} \Gamma\left(\mathscr{c} + \frac{1}{2} \right) - 3 {b_3} {b_2}^3 + \frac{3 {b_2} {b_3}^2 (3^{\mathscr{c}})^6}{2 (\Gamma(\mathscr{c}))^2 \mathscr{c}^2 \pi (2^{\mathscr{c}})^4} \left(\Gamma\left(\mathscr{c} + \frac{1}{3} \right) \right)^2 \\ \left(\Gamma\left(\mathscr{c} + \frac{2}{3} \right) \right)^2 \left(\Gamma\left(\mathscr{c} + \frac{1}{2} \right) \right)^{-2} \end{pmatrix}.$$

$$a_1^{[**]} = \frac{\left[e^{[\kappa]} \right]^{2\mathscr{c}}}{\Gamma(\mathscr{c}+1)} \left(-\frac{\Gamma(3\mathscr{c}+1) b_3}{\Gamma(\mathscr{c}+1) \Gamma(2\mathscr{c}+1)} + \frac{(\Gamma(2\mathscr{c}+1))^2 {b_2}^2}{(\Gamma(\mathscr{c}+1))^4} \right),$$

$$a_2^{[**]} = \frac{\left[e^{[\kappa]} \right]^{3\mathscr{c}}}{\Gamma(\mathscr{c}+1)} \left(\frac{b_2 \Gamma(3\mathscr{c}+1) b_3}{(\Gamma(\mathscr{c}+1))^3} - \frac{\left((\Gamma(2\mathscr{c}+1))^3 {b_2}^2 - \Gamma(3\mathscr{c}+1) b_3 (\Gamma(\mathscr{c}+1))^3 \right) b_2}{(\Gamma(\mathscr{c}+1))^6} \right),$$

$$a_3^{[**]} = \frac{\left[e^{[\kappa]} \right]^{3\mathscr{c}}}{\Gamma(\mathscr{c}+1)} \left(\frac{b_2 \Gamma(3\mathscr{c}+1) b_3}{(\Gamma(\mathscr{c}+1))^3} - \frac{\left((\Gamma(2\mathscr{c}+1))^3 {b_2}^2 - \Gamma(3\mathscr{c}+1) b_3 (\Gamma(\mathscr{c}+1))^3 \right) b_2}{(\Gamma(\mathscr{c}+1))^6} \right).$$

$$a_4^{[**]} = \left(\frac{b_2}{\Gamma(\mathscr{c}) \mathscr{c}} - \frac{(2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+10/20) b_2}{\mathscr{c}^2 \sqrt{\pi} (\Gamma(\mathscr{c}))^2} \right),$$

$$a_5^{[**]} = \begin{pmatrix} \frac{b_3}{\Gamma(\mathscr{c}) \mathscr{c}} - \frac{(2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+1/2) {b_2}^2}{\mathscr{c}^2 \sqrt{\pi} (\Gamma(\mathscr{c}))^2} - \frac{1}{2} \frac{b_3 (3^{\mathscr{c}})^3 \sqrt{3} \Gamma(\mathscr{c}+10/30) \Gamma(\mathscr{c}+20/30)}{\mathscr{c}^2 \sqrt{\pi} (\Gamma(\mathscr{c}))^2 (2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+10/20)} \\ +\frac{(2.0^{\mathscr{c}})^4 (\Gamma(\mathscr{c}+10/20))^2 {b_2}^2}{(\Gamma(\mathscr{c}))^3 \mathscr{c}^3 \pi} \end{pmatrix},$$

$$a_6^{[**]} = \begin{pmatrix} -b_3 + \frac{(2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+1/2) {b_2}^2}{\Gamma(\mathscr{c}) \mathscr{c} \sqrt{\pi}} + \frac{10}{20} \frac{b_3 (3^{\mathscr{c}})^3 \sqrt{3} \Gamma(\mathscr{c}+10/30) \Gamma(\mathscr{c}+20/30)}{\Gamma(\mathscr{c}) \mathscr{c} \sqrt{\pi} (2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+1/2)} \\ -\frac{(2.0^{\mathscr{c}})^4 (\Gamma(\mathscr{c}+10/20))^2 {b_2}^2}{(\Gamma(\mathscr{c}))^2 \mathscr{c}^2 \pi} \end{pmatrix}.$$

$$a^{\{1\}} = a^{\{1\}} = \left(-\frac{b_2 \left(\Gamma(\mathscr{c}) \mathscr{c} \sqrt{\pi} - (2^{\mathscr{c}})^2 \Gamma(\mathscr{c}+10/20) \right)}{\Gamma(\mathscr{c}) \mathscr{c} \sqrt{\pi}} \right),$$

$$a^{\{2\}} = \left(\frac{\left(2 b_3 (\Gamma(\mathscr{c}))^2 \mathscr{c}^2 \pi^{3/2} (2.0^{\mathscr{c}})^2 \Gamma(\mathscr{c}+10/20) - 2 (2^{\mathscr{c}})^4 (\Gamma(\mathscr{c}+10/20))^2 {b_2}^2 \right)}{a^{\{5\}} = \left((\Gamma(\mathscr{c}))^2 \mathscr{c}^2 \pi^{3/2} (2.0^{\mathscr{c}})^2 \Gamma(\mathscr{c}+1/2) \right)} \right).$$

https://doi.org/10.3390/mca31020040

$$a_1^{[*]} = \left(-b_2 + \frac{(2^{\mathscr{c}})^2 \Gamma\left(\mathscr{c} + \frac{10}{20}\right) b_2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} \right),$$

$$a_2^{[*]} = \begin{pmatrix} 1/2\,\frac{b_3(3^{\mathscr{c}})^3\sqrt{3}\Gamma(\mathscr{c}+10/30)\Gamma(\mathscr{c}+20/30)b_2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+10/20)} + \frac{(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+10/20)b_2b_3}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} \\ -\frac{b_3(3^{\mathscr{c}})^3\sqrt{3}\Gamma(\mathscr{c}+1/3)\Gamma(\mathscr{c}+20/30)b_2}{(\Gamma(\mathscr{c}))^2\mathscr{c}^2\pi} - b_4 - \frac{(2^{\mathscr{c}})^4(\Gamma(\mathscr{c}+10/20))^2{b_2}^3}{(\Gamma(\mathscr{c}))^2\mathscr{c}^2\pi} \\ +\frac{{b_2}^3(2^{\mathscr{c}})^6(\Gamma(\mathscr{c}+10/20))^3}{(\Gamma(\mathscr{c}))^3\mathscr{c}^3\pi^{3/2}} \end{pmatrix},$$

$$a_3^{[*]} = \left(2\,a_2^{[*]}b_2 + 3\left(-b_2 + \frac{(2^{\mathscr{c}})^2\Gamma\left(\mathscr{c} + \frac{10}{20}\right)b_2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} \right)^2 b_3 \right).$$

$$a_4^{[*]} = \left(-b_2 + \frac{(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+10/20)b_2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} \right),$$

$$a_5^{[*]} = \begin{pmatrix} 1/2\,\frac{b_3(3^{\mathscr{c}})^3\sqrt{3}\Gamma(\mathscr{c}+1/3)\Gamma(\mathscr{c}+20/30)}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+10/20)} - \\ b_3 + \frac{(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+1/2){b_2}^2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} - \frac{(2^{\mathscr{c}})^4(\Gamma(\mathscr{c}+1/2))^2{b_2}^2}{(\Gamma(\mathscr{c}))^2\mathscr{c}^2\pi} \end{pmatrix},$$

$$a_6^{[*]} = \begin{pmatrix} 1/2\,\frac{b_3(3^{\mathscr{c}})^3\sqrt{3.0}\Gamma(\mathscr{c}+1/3)\Gamma(\mathscr{c}+2/3)b_2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+1/2)} + \frac{(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+10/20)b_2b_3}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} - \\ \frac{b_3(3.0^{\mathscr{c}})^3\sqrt{3}\Gamma(\mathscr{c}+1/3)\Gamma(\mathscr{c}+20/30)b_2}{(\Gamma(\mathscr{c}))^2\mathscr{c}^2\pi} - b_4 - \frac{(2^{\mathscr{c}})^4(\Gamma(\mathscr{c}+10/20))^2{b_2}^3}{(\Gamma(\mathscr{c}))^2\mathscr{c}^2\pi} + \\ \frac{{b_2}^3(2^{\mathscr{c}})^6(\Gamma(\mathscr{c}+10/20))^3}{(\Gamma(\mathscr{c}))^3\mathscr{c}^3\pi^{3/2}} \end{pmatrix},$$

$$a_7^{[*]} = \begin{pmatrix} \left(-b_2 + \frac{(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+10/20)b_2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} \right)^2 b_3 - 4\,\frac{{b_2}^3\left(\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi} - (2^{\mathscr{c}})^2\Gamma(\mathscr{c}+10/20)\right)}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} \\ \left(-b_2 + \frac{(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+10/20)b_2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} \right) \end{pmatrix}.$$

$$a_8^{[*]} = \left(-b_2 + \frac{(2^{\mathscr{c}})^2 b_2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}}\Gamma\left(\mathscr{c} + \frac{1}{2}\right) \right),$$

$$a_9^{[*]} = \begin{pmatrix} \frac{b_3(3^{\mathscr{c}})^3\sqrt{3}}{2\,\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}(2^{\mathscr{c}})^2}\Gamma\left(\mathscr{c} + \frac{1}{3}\right)\Gamma\left(\mathscr{c} + \frac{2}{3}\right) \\ \left(\Gamma\left(\mathscr{c} + \frac{1}{2}\right)\right)^{-1} - b_3 + \frac{(2^{\mathscr{c}})^2{b_2}^2}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}}\Gamma\left(\mathscr{c} + \frac{1}{2}\right) - \frac{(2^{\mathscr{c}})^4{b_2}^2}{(\Gamma(\mathscr{c}))^2\mathscr{c}^2\pi}\left(\Gamma\left(\mathscr{c} + \frac{1}{2}\right)\right)^2 \end{pmatrix},$$

$$a_{10}^{[*]} = \begin{pmatrix} -2\,\frac{(2^{\mathscr{c}})^4(\Gamma(\mathscr{c}+10/20))^2{b_2}^3}{(\Gamma(\mathscr{c}))^2\mathscr{c}^2\pi} + \frac{b_3(3^{\mathscr{c}})^3\sqrt{3}b_2}{2\,\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}(2^{\mathscr{c}})^2}\,\frac{\Gamma\left(\mathscr{c}+\frac{10}{30}\right)\Gamma\left(\mathscr{c}+\frac{20}{30}\right)}{\Gamma\left(\mathscr{c}+\frac{1}{2}\right)} \\ +\frac{(2^{\mathscr{c}})^2b_2b_3}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}}\Gamma\left(\mathscr{c} + \frac{1}{2}\right) + 2.0\,\frac{(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+1/2){b_2}^3}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} + \frac{{b_2}^3(2^{\mathscr{c}})^6}{(\Gamma(\mathscr{c}))^3\mathscr{c}^3\pi^{\frac{3}{2}}}\left(\Gamma\left(\mathscr{c} + \frac{1}{2}\right)\right)^3 \\ -{b_2}^3 - \frac{b_3(3^{\mathscr{c}})^3\sqrt{3}b_2}{(\Gamma(\mathscr{c}))^2\mathscr{c}^2\pi}\Gamma\left(\mathscr{c} + \frac{1}{3}\right)\Gamma\left(\mathscr{c} + \frac{2}{3}\right) - b_4 \end{pmatrix}.$$

$$a_{11}^{[*]} = \left({b_2}^3 - 2\,\frac{(2^{\mathscr{c}})^2\Gamma(\mathscr{c}+10/20){b_2}^3}{\Gamma(\mathscr{c})\mathscr{c}\sqrt{\pi}} + \frac{(2^{\mathscr{c}})^4{b_2}^3}{(\Gamma(\mathscr{c}))^2\mathscr{c}^2\pi}\left(\Gamma\left(\mathscr{c} + \frac{1}{2}\right)\right)^2 \right)$$

$$a_{12}^{[*]} = \begin{pmatrix} 4\,(2^{\mathscr{c}})^6(\Gamma(\mathscr{c}+1/2))^3\Gamma(\mathscr{c})\pi^{3/2}\mathscr{c}\,{b_2}^2 - 2\,(2^{\mathscr{c}})^4(\Gamma(\mathscr{c}+1/2))^2 \\ (\Gamma(\mathscr{c}))^2\pi^2\mathscr{c}^2{b_2}^2 + (2^{\mathscr{c}})^2\Gamma\left(\mathscr{c} + \frac{1}{2}\right)\Gamma(\mathscr{c})\pi^{\frac{3}{2}}(3^{\mathscr{c}})^3 \\ \sqrt{3}\Gamma\left(\mathscr{c} + \frac{1}{3}\right)\Gamma\left(\mathscr{c} + \frac{2}{3}\right)\mathscr{c}\,b_3 - 2\,(2^{\mathscr{c}})^8\left(\Gamma\left(\mathscr{c} + \frac{1}{2}\right)\right)^4\pi\,{b_2}^2 + \\ 2\,(2^{\mathscr{c}})^2\Gamma\left(\mathscr{c} + \frac{1}{2}\right)(\Gamma(\mathscr{c}))^3\pi^{5/2}\mathscr{c}^3b_3 - \end{pmatrix},$$

$$a_{13}^{[*]} = \begin{pmatrix} (2^{\mathscr{C}})^4 \left(\Gamma\left(\mathscr{C}+\frac{10}{20}\right)\right)^2 \sqrt{\pi} - \\ 2\,(2.0^{\mathscr{C}})^2 \Gamma\left(\mathscr{C}+\frac{10}{20}\right)\Gamma(\mathscr{C})\pi\,\mathscr{C} + (\Gamma(\mathscr{C}))^2 \mathscr{C}^2 \pi^{\frac{3}{2}} \end{pmatrix}.$$

$$a_{14}^{[*]} = \left(-b_2 + \frac{(2^{\mathscr{C}})^2 b_2}{\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}}\Gamma\left(\mathscr{C}+\frac{1}{2}\right)\right),$$

$$a_{15}^{[*]} = \begin{pmatrix} \frac{b_3 (3^{\mathscr{C}})^3 \sqrt{3}}{2.0\,\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}(2^{\mathscr{C}})^2} \frac{\Gamma\left(\mathscr{C}+\frac{10}{30}\right)\Gamma\left(\mathscr{C}+\frac{20}{30}\right)}{\Gamma\left(\mathscr{C}+\frac{1}{2}\right)} - \\ b_3 + \frac{(2^{\mathscr{C}})^2 {b_2}^2}{\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}}\Gamma\left(\mathscr{C}+\frac{1}{2}\right) - \frac{(2^{\mathscr{C}})^4 {b_2}^2}{(\Gamma(\mathscr{C}))^2 \mathscr{C}^2 \pi}\left(\Gamma\left(\mathscr{C}+\frac{1}{2}\right)\right)^2 \end{pmatrix},$$

$$a_{16}^{[*]} = \begin{pmatrix} \frac{(2^{\mathscr{C}})^2 b_2 b_3}{\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}}\Gamma\left(\mathscr{C}+\frac{1}{2}\right) + \frac{b_3 (3^{\mathscr{C}})^3 \sqrt{3} b_2}{2\,\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}(2^{\mathscr{C}})^2} \frac{\Gamma\left(\mathscr{C}+\frac{10}{30}\right)\Gamma\left(\mathscr{C}+\frac{2}{3}\right)}{\Gamma\left(\mathscr{C}+\frac{10}{20}\right)} - b_4 + 2\,{b_2}^3 \\ + \frac{{b_2}^3 (2^{\mathscr{C}})^6}{(\Gamma(\mathscr{C}))^3 \mathscr{C}^3 \pi^{\frac{3}{2}}}\left(\Gamma\left(\mathscr{C}+\frac{1}{2}\right)\right)^3 - 4\,\frac{(2^{\mathscr{C}})^2 \Gamma(\mathscr{C}+1/2) {b_2}^3}{\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}} + \\ \frac{(2.0^{\mathscr{C}})^4 {b_2}^3}{(\Gamma(\mathscr{C}))^2 \mathscr{C}^2 \pi}\left(\Gamma\left(\mathscr{C}+\frac{1}{2}\right)\right)^2 - \frac{b_3 (3^{\mathscr{C}})^3 \sqrt{3} b_2}{(\Gamma(\mathscr{C}))^2 \mathscr{C}^2 \pi}\Gamma\left(\mathscr{C}+\frac{10}{30}\right)\Gamma\left(\mathscr{C}+\frac{2}{3}\right) \end{pmatrix}.$$

$$a_{17}^{[*]} = \left(-b_2 + \frac{(2^{\mathscr{C}})^2 b_2}{\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}}\Gamma\left(\mathscr{C}+\frac{1}{2}\right)\right),$$

$$a_{18}^{[*]} = \begin{pmatrix} \frac{b_3 (3^{\mathscr{C}})^3 \sqrt{3}}{2\,\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}(2^{\mathscr{C}})^2}\Gamma\left(\mathscr{C}+\frac{1}{3}\right)\Gamma\left(\mathscr{C}+\frac{2}{3}\right)\left(\Gamma\left(\mathscr{C}+\frac{1}{2}\right)\right)^{-1} - b_3 \\ + \frac{(2^{\mathscr{C}})^2 {b_2}^2}{\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}}\Gamma\left(\mathscr{C}+\frac{1}{2}\right) - \frac{(2^{\mathscr{C}})^4 {b_2}^2}{(\Gamma(\mathscr{C}))^2 \mathscr{C}^2 \pi}\left(\Gamma\left(\mathscr{C}+\frac{1}{2}\right)\right)^2 \end{pmatrix},$$

$$a_{19}^{[*]} = \begin{pmatrix} \frac{b_3 (3.0^{\mathscr{C}})^3 \sqrt{3} b_2}{2\,\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}(2^{\mathscr{C}})^2} \frac{\Gamma\left(\mathscr{C}+\frac{10}{30}\right)\Gamma\left(\mathscr{C}+\frac{2}{3}\right)}{\Gamma\left(\mathscr{C}+\frac{1}{2}\right)} + \frac{(2.0^{\mathscr{C}})^2 b_2 b_3}{\Gamma(\mathscr{C})\mathscr{C}\sqrt{\pi}}\Gamma\left(\mathscr{C}+\frac{1}{2}\right) \\ -\frac{b_3 (3^{\mathscr{C}})^3 \sqrt{3.0} b_2}{(\Gamma(\mathscr{C}))^2 \mathscr{C}^2 \pi}\Gamma\left(\mathscr{C}+\frac{1}{3}\right)\Gamma\left(\mathscr{C}+\frac{2}{3}\right) - b_4 - \frac{(2^{\mathscr{C}})^4 {b_2}^3}{(\Gamma(\mathscr{C}))^2 \mathscr{C}^2 \pi}\left(\Gamma\left(\mathscr{C}+\frac{1}{2}\right)\right)^2 \\ + \frac{{b_2}^3 (2.0^{\mathscr{C}})^6}{(\Gamma(\mathscr{C}))^3 \mathscr{C}^3 \pi^{\frac{30}{20}}}\left(\Gamma\left(\mathscr{C}+\frac{10}{20}\right)\right)^3 \end{pmatrix}.$$

References

1. Thirthar, A.A.; Panja, P.; Khan, A.; Alqudah, M.A.; Abdeljawad, T. An ecosystem model with memory effect considering global warming phenomena and an exponential fear function. *Fractals* **2023**, *31*, 2340162. [CrossRef]
2. Wang, L.; Liu, F.; Fang, Y.; Ma, J.; Wang, J.; Qu, L.; Yang, Q.; Wu, W.; Jin, L.; Sun, D. Advances in zebrafish as a comprehensive model of mental disorders. *Depress. Anxiety* **2023**, *2023*, 6663141. [CrossRef]
3. Ahmad, I.; Jan, R.; Razak, N.N.A.; Khan, A.; Abdeljawad, T. Exploring Fractional-Order Models in Computational Finance via an Efficient Hybrid Approach. *Eur. J. Pure Appl. Math.* **2025**, *18*, 5793. [CrossRef]
4. Rebizant, W.; Szafran, J.; Wiszniewski, A. *Digital Signal Processing in Power System Protection and Control*; Springer Science and Business Media: Berlin/Heidelberg, Germany, 2011.
5. Ahmad, I.; Jan, R.; Razak, N.N.A.; Khan, A.; Abdeljawad, T. Computational Study of a Meshless Approach for Multi-Term Time-Fractional Models in Drug Dispersion and Absorption in Biological Tissues. *Eur. J. Pure Appl. Math.* **2025**, *18*, 5684. [CrossRef]
6. Ahmad, I.; Mekawy, I.; Khan, M.N.; Jan, R.; Boulaaras, S. Modeling anomalous transport in fractal porous media: A study of fractional diffusion PDEs using numerical method. *Nonlinear Eng.* **2024**, *13*, 20220366. [CrossRef]
7. Roylance, D. *Engineering Viscoelasticity*; Department of Materials Science and Engineering Massachusetts Institute of Technology: Cambridge, MA, USA, 2001; pp. 1–37.
8. Asgharian, H.; Baniasadi, E. A review on modeling and simulation of solar energy storage systems based on phase change materials. *J. Energy Storage* **2019**, *21*, 186–201. [CrossRef]
9. Shahzad, M.; Ahmed, N.; Iqbal, M.S.; Inc, M.; Baber, M.Z.; Anjum, R.; Shahid, N. Application of Fixed Point Theory and Solitary Wave Solutions for the Time-Fractional Nonlinear Unsteady Convection-Diffusion System. *Int. J. Theor. Phys.* **2023**, *62*, 263. [CrossRef]
10. Sahimi, M. Non-linear and non-local transport processes in heterogeneous media: From long-range correlated percolation to fracture and materials breakdown. *Phys. Rep.* **1998**, *306*, 213–395. [CrossRef]
11. Charkaoui, A.; Ben-loghfyry, A. Anisotropic equation based on fractional diffusion tensor for image noise removal. *Math. Methods Appl. Sci.* **2024**, *47*, 9600–9620. [CrossRef]

12. Jan, R.; Ahmad, I.; Ahmad, H.; Vrinceanu, N.; Hasegan, A.G. Insights into dengue transmission modeling: Index of memory, carriers, and vaccination dynamics explored via non-integer derivative. *AIMS Bioeng.* **2024**, *11*, 44–65. [CrossRef]
13. Ben-loghfyry, A.; Hakim, A. Reaction-diffusion equation based on fractional-time anisotropic diffusion for textured images recovery. *Int. J. Appl. Comput. Math.* **2022**, *8*, 177. [CrossRef]
14. Ben-Loghfyry, A.; Charkaoui, A. A novel Perona-Malik model driven by fractional regularized diffusion mechanism with caputo derivative applied to magnetic resonance image processing. *J. Frankl. Inst.* **2025**, *362*, 107950. [CrossRef]
15. Charkaoui, A.; Ben-Loghfyry, A. A class of nonlinear parabolic PDEs with variable growth structure applied to multi-frame MRI super-resolution. *Nonlinear Anal. Real World Appl.* **2025**, *83*, 104259. [CrossRef]
16. Wassef, M.; Luscan, A.; Battistella, A.; Le Corre, S.; Li, H.; Wallace, M.R.; Vidaud, M.; Margueron, R. Versatile and precise gene-targeting strategies for functional studies in mammalian cell lines. *Methods* **2017**, *121*, 45–54. [CrossRef]
17. Fujimoto, R.; Bock, C.; Chen, W.; Page, E.; Panchal, J.H. (Eds.) *Research Challenges in Modeling and Simulation for Engineering Complex Systems*; Springer International Publishing: Cham, Switzerland, 2017.
18. Ahmed, N.; Baber, M.Z.; Iqbal, M.S.; Akgül, A.; Rafiq, M.; Raza, A.; Chowdhury, M.S.R. Investigation of soliton structures for dispersion, dissipation, and reaction time-fractional KdV–Burgers–Fisher equation with the noise effect. *Int. J. Model. Simul.* **2025**, *45*, 2199–2215. [CrossRef]
19. Ben-Loghfyry, A.; Hakim, A. A novel robust fractional-time anisotropic diffusion for multi-frame image super-resolution. *Adv. Comput. Math.* **2023**, *49*, 79. [CrossRef]
20. Bayrak, M.A.; Demir, A.; Ozbilge, E. On Fractional Newton-Type Method for Nonlinear Problems. *J. Math.* **2022**, *2022*, 7070253. [CrossRef]
21. Torres-Hernandez, A.; Brambila-Paz, F. Fractional Newton-Raphson method and some variants for the solution of non-linear systems. *arXiv* **2019**, arXiv:1908.01453.
22. Chew, J.V.L.; Sunarto, A.; Sulaiman, J.; Asli, M.F. Fractional Newton explicit group method for time-fractional nonlinear porous medium equations. *Prog. Fract. Differ. Appl.* **2024**, *10*, 391–398. [CrossRef]
23. Liu, C.S.; Chang, C.W. Updating to Optimal Parametric Values by Memory-Dependent Methods: Iterative Schemes of Fractional Type for Solving Nonlinear Equations. *Mathematics* **2024**, *12*, 1032. [CrossRef]
24. Singh, H.; Sharma, J.R. A fractional Traub-Steffensen-type method for solving nonlinear equations. *Numer. Algorithms* **2024**, *95*, 1103–1126. [CrossRef]
25. Gdawiec, K.; Kotarski, W.; Lisowska, A. Visual analysis of the Newton's method with fractional order derivatives. *Symmetry* **2019**, *11*, 1143. [CrossRef]
26. Candelario, G.; Cordero, A.; Torregrosa, J.R.; Vassileva, M.P. Generalized conformable fractional Newton-type method for solving nonlinear systems. *Numer. Algorithms* **2023**, *93*, 1171–1208. [CrossRef]
27. Almeida, R. A Caputo fractional derivative of a function with respect to another function. *Commun. Nonlinear Sci. Numer. Simul.* **2017**, *44*, 460–481. [CrossRef]
28. Gao, G.H.; Sun, Z.Z.; Zhang, H.W. A new fractional numerical differentiation formula to approximate the Caputo fractional derivative and its applications. *J. Comput. Phys.* **2014**, *259*, 33–50. [CrossRef]
29. Benjemaa, M. Taylor's formula involving generalized fractional derivatives. *Appl. Math. Comput.* **2018**, *335*, 182–195. [CrossRef]
30. Shahzad, M.; Ahmed, N.; Iqbal, M.S.; Inc, M.; Baber, M.Z.; Anjum, R. Classical Regularity and Wave Structures of Fractional Order Selkov-Schnakenberg System. *Int. J. Theor. Phys.* **2024**, *63*, 95. [CrossRef]
31. Torres-Hernandez, A.; Brambila-Paz, F. Sets of fractional operators and numerical estimation of the order of convergence of a family of fractional fixed-point methods. *Fractal Fract.* **2021**, *5*, 240. [CrossRef]
32. Shams, M.; Kausar, N.; Agarwal, P.; Jain, S.; Salman, M.A.; Shah, M.A. On family of the Caputo-type fractional numerical scheme for solving polynomial equations. *Appl. Math. Sci. Eng.* **2023**, *31*, 2181959. [CrossRef]
33. Ali, N.; Waseem, M.; Safdar, M.; Akgul, A.; Tolasa, F.T. Iterative solutions for nonlinear equations via fractional derivatives: Adaptations and advances. *Appl. Math. Sci. Eng.* **2024**, *32*, 2333816. [CrossRef]
34. Akgul, A.; Cordero, A.; Torregrosa, J.R. A fractional Newton method with 2αth-order of convergence and its stability. *Appl. Math. Lett.* **2019**, *98*, 344–351. [CrossRef]
35. Candelario, G.; Cordero, A.; Torregrosa, J.R. Multipoint fractional iterative methods with $(2\alpha + 1)$ th-order of convergence for solving nonlinear problems. *Mathematics* **2020**, *8*, 452. [CrossRef]
36. Behl, R.; Salimi, M.; Ferrara, M.; Sharifi, S.; Alharbi, S.K. Some real-life applications of a newly constructed derivative free iterative scheme. *Symmetry* **2019**, *11*, 239. [CrossRef]
37. Behl, R.; Bhalla, S.; Chun, C. Two-step iterative methods for multiple roots and their applications for solving several physical and chemical problems. *Math. Methods Appl. Sci.* **2023**, *46*, 8877–8894. [CrossRef]
38. Shams, M.; Carpentieri, B. An efficient and stable caputo-type inverse fractional parallel scheme for solving nonlinear equations. *Axioms* **2024**, *13*, 671. [CrossRef]

39. Ahmadi, A.A.; Majumdar, A. Some applications of polynomial optimization in operations research and real-time decision making. *Optim. Lett.* **2016**, *10*, 709–729. [CrossRef]
40. Pourhejazy, P. Destruction decisions for managing excess inventory in e-commerce logistics. *Sustainability* **2020**, *12*, 8365. [CrossRef]