

Article

Robust Fractional Quantum Two-Step Schemes with Enhanced Stability for Nonlinear Equations

Mudassir Shams ^{1,2} and Bruno Carpentieri ^{3,*} ¹ Department of Mathematics, Faculty of Arts and Science, Balikesir University, Balikesir 10145, Turkey; mudassir.shams@balikesir.edu.tr² Department of Mathematics and Statistics, Riphah International University, I-14, Islamabad 44000, Pakistan³ Faculty of Engineering, Free University of Bozen-Bolzano, 39100 Bolzano, Italy

* Correspondence: bruno.carpentieri@unibz.it

Abstract

Fractional quantum calculus provides a powerful mathematical framework for incorporating memory and scaling effects into numerical models. However, classical iterative methods for nonlinear equations often suffer from limited stability, sensitivity to initial guesses, and restricted convergence domains, particularly in highly nonlinear settings. In this work, we introduce a new Caputo fractional–quantum iterative scheme, denoted by $MSB_{q;\alpha}$, formulated as a parameterized two-step method based on a Caputo-type fractional quantum derivative. The proposed framework incorporates additional structural parameters that regulate the iterative dynamics and enable enhanced control over convergence behavior and stability properties. To assess the performance of the method, we employ tools from complex dynamical systems, including stability analysis and fractal basin investigations in the complex plane. These analyses provide insight into how the fractional and quantum parameters influence the geometry of attraction domains and the global convergence behavior of the scheme. Numerical experiments on representative nonlinear problems arising in engineering and biomedical applications demonstrate improved robustness with respect to initial guesses, reduced residual errors, and competitive computational efficiency compared with existing iterative methods. Overall, the results indicate that the proposed fractional–quantum framework offers an effective and versatile approach for the numerical solution of challenging nonlinear equations.

Keywords: fractional quantum calculus; two-step iterative methods; nonlinear equations; dynamical systems analysis; convergence acceleration; engineering and biomedical applications

MSC: 65H05; 26A33; 39A13; 65B99; 37M99



Academic Editor: Churong Chen

Received: 20 March 2026

Revised: 14 April 2026

Accepted: 21 April 2026

Published: 27 April 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Nonlinear equations constitute a fundamental mathematical framework for modeling equilibrium states, steady configurations, and implicit relationships arising in science and engineering. Given a nonlinear operator

$$f(x) = 0, \quad f : \Omega \subset \mathbb{R} \rightarrow \mathbb{R}, \quad (1)$$

the objective is to determine $x^* \in \Omega$ such that $f(x^*) = 0$. This abstract formulation arises in numerous contexts, including steady states of differential systems, discretized

fractional partial differential equations, nonlinear inverse problems, and parameter identification models. In practice, the structural properties of f , such as smoothness, nonlocality, and memory dependence, strongly influence the performance and stability of numerical solvers designed to approximate its roots [1–4].

Classical calculus assumes locality; that is, the derivative at a point depends solely on infinitesimal neighborhood information. However, many real-world systems exhibit hereditary effects, long-range interactions, and scaling symmetries that cannot be adequately captured within this local framework. Fractional calculus [5] generalizes differentiation to non-integer orders, thereby incorporating memory effects through integral operators; for example, the Caputo derivative of order $\alpha \in (0, 1)$ introduces nonlocal dependence via a power-law kernel, allowing past states of a system to influence its present behavior. Such operators arise naturally in models of viscoelasticity, anomalous diffusion, bioheat transfer, and population dynamics with memory.

Parallel to fractional calculus, quantum calculus (or q -calculus) eliminates the classical limit process and replaces differentiation with finite scaling differences [6]. The Jackson q -derivative is defined as

$$D_q f(x) = \frac{f(qx) - f(x)}{(q-1)x}, \quad q \neq 1, \quad (2)$$

and captures discrete self-similarity and multiplicative scaling effects. Fractional quantum operators combine these two extensions, enabling the simultaneous representation of memory and scaling phenomena [7,8]. From a theoretical perspective, these operators enlarge the functional space in which nonlinear operators act and provide additional structural flexibility for constructing iterative numerical schemes.

Root-finding methods are typically formulated as fixed-point iterations,

$$x_{k+1} = G(x_k), \quad (3)$$

where $G: \Omega \rightarrow \Omega$ satisfies $G(x^*) = x^*$. The convergence of (3) is governed by the spectral radius of the derivative,

$$\rho(G'(x^*)) < 1.$$

Among classical approaches, Newton's method

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \quad (4)$$

is widely used due to its quadratic convergence under standard regularity assumptions. However, its convergence guarantees are essentially local, and the size of the basin of attraction depends strongly on the nonlinear structure of f . For highly nonlinear or stiff operators, the associated iterative map may exhibit multiple attracting cycles, chaotic regions, or narrow attraction domains [9].

From the perspective of complex dynamics [10], iterative schemes generate intricate fractal basin structures in the complex plane. Such analyses provide valuable insight into the global stability properties of iterative methods beyond classical convergence order. Consequently, the convergence behavior of a method depends not only on its local order but also on the global dynamics of the iteration map G . This observation motivates the development of iterative frameworks with additional structural parameters capable of reshaping attraction domains and improving global convergence behavior. In this context, fractional and quantum calculus provide promising tools for enhancing both the flexibility and stability of iterative numerical schemes. Building on these observations, several studies have extended classical iterative methods by incorporating fractional and q -calculus

operators to improve convergence, stability, and global dynamical behavior. For instance, Ali et al. investigated iterative schemes based on fractional derivatives and reported improved efficiency compared with classical root-finding methods [11]. Similarly, Sana et al. developed q -iterative techniques derived from q -Taylor expansions and coupled system formulations, demonstrating enhanced convergence properties [12]. Other q -analogue approaches have also been proposed, achieving higher convergence rates under suitable choices of the parameter q [13,14]. Moreover, multi-step and parallel q -fractional algorithms have been introduced to approximate multiple roots simultaneously, while providing insight into the global convergence behavior of the associated iterative maps [15].

Despite these advances, a unified framework that systematically integrates fractional and quantum operators to control the geometry of attraction domains and the dynamical properties of iterative schemes remains largely unexplored. More specifically, while fractional Newton-type methods [16,17] and q -deformed schemes [18] have been investigated separately in the literature, several theoretical and computational questions remain open, as detailed below.

- The absence of a unified framework combining Caputo-type memory operators with quantum scaling derivatives within multi-step iterative structures.
- Limited theoretical understanding of how fractional and quantum parameters influence the dynamical system generated by the iteration map.
- Insufficient exploration of parameterized iterative families as mechanisms for controlling stability regions and convergence behavior.
- A lack of systematic investigation of the geometry of attraction basins under fractional–quantum perturbations.

In particular, the combined integration of fractional and quantum operators within a single iterative framework—and the resulting impact on convergence and stability properties—has not yet been systematically investigated.

Existing approaches (see [19–21] and the references therein) typically treat the fractional order α and the quantum parameter q as fixed constants, rather than as intrinsic regulators of the convergence dynamics. In contrast, we introduce a parameterized family of two-step fractional quantum iterative schemes in which these quantities, together with additional structural parameters, actively shape the iterative process. The proposed formulation combines fractional and quantum operators within a two-step structure that enables enhanced control of accuracy, stability, and convergence behavior.

The resulting formulation defines a family of iterative maps whose fixed points correspond to solutions of the nonlinear equation. This perspective provides a natural and rigorous basis for both convergence analysis and the investigation of fractal basin structures associated with the dynamical behavior of the proposed schemes. Accordingly, the proposed approach offers a unified viewpoint for the design and analysis of fractional–quantum iterative methods. The principal theoretical contributions of this work can be summarized as follows:

1. Formulation of a unified fractional–quantum operator suitable for the construction of iterative root-finding schemes.
2. Development of a parameterized two-step iterative framework providing additional structural flexibility for controlling convergence behavior.
3. Derivation of local convergence properties via a perturbation analysis of the classical Taylor expansion in the presence of fractional operators.
4. Dynamical systems analysis of the proposed schemes, including the study of stability and basin geometry in the complex plane.

5. Identification of fractional and quantum parameters as geometric regulators capable of modifying the structure of attraction domains.

A key aspect of the proposed approach is the interpretation of fractional and quantum operators not merely as modeling tools, but as intrinsic components of iterative dynamics. By embedding Caputo-type memory effects and q -scaling operators directly into the iterative structure, we construct a dynamical formulation whose convergence behavior can be continuously tuned through the parameters $(\alpha, q, \beta_1, \beta_2)$, where β_1 and β_2 are additional structural parameters in the proposed schemes.

It is important to emphasize the role of the fractional-order parameter $\alpha \in (0, 1]$. The case $\alpha = 1$ corresponds to the classical integer-order model and serves as a benchmark for validation. In contrast, $\alpha < 1$ enables the incorporation of memory and hereditary effects, which are essential for modeling complex dynamical systems with nonlocal behavior. Thus, the fractional formulation provides enhanced modeling flexibility, even though certain numerical schemes may attain optimal convergence in the limiting case $\alpha = 1$.

To the best of our knowledge, the systematic combination of the following elements has not yet been explored within a unified formulation for solving nonlinear equations. While existing numerical approaches for Caputo fractional problems typically treat fractional operators as external modeling components, the present work integrates them directly into the iterative structure, thereby enhancing both flexibility and convergence behavior.

The unified formulation proposed in this paper enables a more comprehensive analysis of nonlinear problems by combining memory effects, quantum scaling, and iterative dynamics within a single framework, thereby distinguishing the proposed method from existing numerical approaches in the following aspects:

- Caputo-type fractional derivatives,
- quantum (q -) difference operators,
- parameterized two-step iterative schemes,
- complex fractal basin analysis.

The remainder of this paper is organized as follows. Section 2 introduces the fractional–quantum operators and establishes their main analytical properties. Section 3 develops the proposed family of two-step iterative schemes and presents the corresponding convergence analysis. Section 4 investigates the stability and dynamical behavior of the methods, including the analysis of fractal basin structures. Section 5 presents numerical experiments on nonlinear problems arising in engineering and biomedical applications. Finally, Section 6 concludes the paper and outlines directions for future research.

2. Preliminaries

In this section, we recall the main concepts of q -calculus and fractional q -calculus that form the analytical foundation for the fractional q -iterative schemes developed later in the paper. Throughout the paper, we assume $q \in (0, 1)$ and consider $0 < a < x < b$. We denote by $\mathbb{N} = \{1, 2, \dots\}$ the set of positive integers and by $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$ the set of non-negative integers.

2.1. Basic Elements of q -Calculus

The concept of q -analogues extends classical algebraic quantities by introducing a deformation parameter q . These generalized quantities reduce to their classical counterparts in the limit $q \rightarrow 1^-$.

Definition 1 (q-Number and q-Factorial [22]). The q-number associated with $r \in \mathbb{R}$ is defined by

$$[r]_q = \frac{1 - q^r}{1 - q}. \quad (5)$$

For $n \in \mathbb{N}$, the q-factorial is defined recursively as

$$[n]_q! = \prod_{k=1}^n [k]_q, \quad [0]_q! = 1. \quad (6)$$

The q-shifted power plays a fundamental role in the construction of fractional kernels and related operators.

Definition 2 (q-Shifted Power [23]). For $a, b \in \mathbb{R}$ and $n \in \mathbb{N}$,

$$(a - b)_q^n = \prod_{j=0}^{n-1} (a - bq^j). \quad (7)$$

More generally, for $\theta \in \mathbb{R}$,

$$(a - b)_q^\theta = a^\theta \prod_{j=0}^{\infty} \frac{a - bq^j}{a - bq^{j+\theta}}, \quad (8)$$

whenever the infinite product converges.

The central operator in quantum calculus is the Jackson q-derivative, which replaces the classical limit process with multiplicative scaling.

Definition 3 (Jackson q-Derivative [24]). Let $f : [0, b] \rightarrow \mathbb{R}$. The Jackson q-derivative is defined by

$$(D_q f)(x) = \frac{f(qx) - f(x)}{(q - 1)x}, \quad x \neq 0, \quad (9)$$

with

$$(D_q f)(0) = \lim_{x \rightarrow 0} (D_q f)(x), \quad (10)$$

provided the limit exists. Higher-order derivatives are defined recursively as

$$D_q^n f := D_q(D_q^{n-1} f), \quad n \in \mathbb{N}. \quad (11)$$

2.2. Jackson Integral and q-Gamma Function

The integral counterpart of the Jackson q-derivative is the Jackson q-integral, which establishes an inversion relationship analogous to the fundamental theorem of classical calculus.

Definition 4 (Jackson q-Integral [25]). The Jackson q-integral over $[0, x]$ is defined by

$$\int_0^x f(t) d_q t = (1 - q)x \sum_{k=0}^{\infty} q^k f(xq^k), \quad (12)$$

whenever the series converges.

The following identity expresses the inverse relation between the q-derivative and the Jackson q-integral.

Proposition 1 (Fundamental Theorem of q -Calculus [26]). *If f is continuous at 0, then*

$$D_q \left(\int_0^x f(t) d_q t \right) = f(x). \quad (13)$$

To extend q -calculus operators to fractional orders, the q -Gamma function is introduced as a normalization factor.

Definition 5 (q -Gamma Function [27]). *For $\alpha > 0$, the q -Gamma function is defined by*

$$\Gamma_q(\alpha) = (1 - q)^{1-\alpha} \frac{(q; q)_\infty}{(q^\alpha; q)_\infty}, \quad (14)$$

where

$$(a; q)_\infty = \prod_{k=0}^{\infty} (1 - aq^k). \quad (15)$$

It satisfies the functional relation

$$\Gamma_q(\alpha + 1) = [\alpha]_q \Gamma_q(\alpha). \quad (16)$$

2.3. Fractional q -Operators

Fractional q -operators extend the concepts of q -calculus to non-integer orders and provide the analytical framework required for the construction of fractional q -iterative schemes.

Definition 6 (Riemann–Liouville Fractional q -Integral [28]). *For $\alpha > 0$, the Riemann–Liouville fractional q -integral of order α is defined by*

$$(I_{q,a}^{[\alpha]} f)(x) = \frac{1}{\Gamma_q(\alpha)} \int_a^x (x - t)_q^{\alpha-1} f(t) d_q t. \quad (17)$$

Based on this operator, the corresponding Riemann–Liouville fractional q -derivative is defined as follows.

Definition 7 (Riemann–Liouville Fractional q -Derivative [29]). *Let $\alpha > 0$ and $n = \lceil \alpha \rceil$. Then*

$$(D_{q,a}^{[\alpha]} f)(x) = D_q^n (I_{q,a}^{n-\alpha} f)(x). \quad (18)$$

An alternative formulation frequently used in applications is the Caputo-type fractional q -derivative, which differs from the Riemann–Liouville derivative in the order of differentiation and integration.

Definition 8 (Caputo Fractional q -Derivative [30]). *Let $\alpha > 0$ and $n = \lceil \alpha \rceil$. Then*

$$({}^C D_{q,a}^{[\alpha]} f)(x) = I_{q,a}^{n-\alpha} (D_q^n f)(x). \quad (19)$$

2.4. Key Properties and Fractional q -Taylor Formula

The following identities describe the interaction between classical q -derivatives and fractional q -operators introduced in the previous subsection.

Proposition 2. *For $\alpha \in \mathbb{R}^+ \setminus \mathbb{N}$ and $0 < a < x < b$,*

$${}^C D_{q,a}^{[\alpha]} (D_q f)(x) = {}^C D_{q,a}^{[\alpha+1]} f(x), \quad (20)$$

and

$$D_q \left({}^C D_{q,a}^{[\alpha]} f \right) (x) = {}^C D_{q,a}^{[\alpha+1]} f(x) + \frac{D_q^{[\alpha]} f(a)}{\Gamma_q([\alpha] - \alpha)} (x - a)_q^{[\alpha] - \alpha - 1}. \quad (21)$$

The fractional q -Taylor expansion provides the fundamental analytical tool for deriving local error equations in fractional q -iterative methods.

Theorem 1 (Fractional q -Taylor Expansion [31]). *Let f be sufficiently smooth and $\alpha > 0$. Then*

$$f(x) = \sum_{k=0}^{n-1} \frac{(x-a)_q^k}{[k]_q!} D_q^k f(a) + \frac{1}{\Gamma_q(\alpha)} \int_a^x (x-t)_q^{\alpha-1} ({}^C D_{q,a}^{[\alpha]} f)(t) d_q t, \quad (22)$$

where $n = \lceil \alpha \rceil$.

3. Development and Analysis of the q -Fractional Scheme

3.1. Proposed Fractional Weighted Iterative Method

Let ξ be a simple root of $f : \mathbb{R} \rightarrow \mathbb{R}$, and define the error at the k -th iteration by

$$e_k := x_k - \xi.$$

We now introduce the proposed two-step weighted iterative method, abbreviated as $MSB_{q;\alpha}$, for the solution of nonlinear equations:

$$\begin{cases} y_k = x_k - \left(\Gamma_q(\alpha + 1) \frac{f(x_k)}{[{}^C D_q^{[\alpha]} f](x_k)} \left(\frac{1}{1 - \frac{\beta_1 f(x_k)}{1 + \beta_2 f(x_k)}} - f(x_k) \right) \right)^{\frac{1}{\alpha}}, \\ x_{k+1} = y_k - \left(\Gamma_q(\alpha + 1) \frac{f(x_k)}{[{}^C D_q^{[\alpha]} f](x_k)} \left(\frac{f(y_k)}{f(x_k)} + 2 \left(\frac{f(y_k)}{f(x_k)} \right)^2 \right) \right)^{\frac{1}{\alpha}}. \end{cases} \quad (23)$$

Here, $\beta_1, \beta_2 \in \mathbb{R}$ are free parameters that regulate the convergence behavior of the scheme.

Well-Posedness of the Fractional Power and Branch Selection

The fractional power $(\cdot)^{1/\alpha}$ appearing in the iterative scheme (23) is defined via the principal branch of the complex logarithm:

$$z^{1/\alpha} := \exp\left(\frac{1}{\alpha} \log z\right),$$

where $\log z$ denotes the principal branch of the complex logarithm.

For real-valued problems, to preserve real iterates, we employ the sign-preserving definition

$$z^{1/\alpha} := \text{sign}(z) |z|^{1/\alpha}, \quad z \in \mathbb{R}.$$

Moreover, for $\alpha \in (0, 1)$, the well-posedness of the proposed scheme is ensured under standard assumptions. Specifically, if f is sufficiently smooth in a neighborhood of the simple root ξ and the initial guess is sufficiently close to ξ , then the iteration is well-defined and locally convergent. In this setting, existence and uniqueness of the root are guaranteed, and the iterates depend continuously on the initial guess, ensuring stability in the sense of Hadamard.

Under these conditions, the argument of the fractional power remains positive in a neighborhood of the root, and hence the iterates remain real.

Theorem 2 (Preservation of Real Iterates). *Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be sufficiently smooth, and let $\xi \in \mathbb{R}$ be a simple root of f , i.e., $f(\xi) = 0$ and ${}^C D_q^{[\alpha]} f(\xi) \neq 0$.*

Assume that the fractional iterative scheme (23) is applied with an initial guess $x_0 \in \mathbb{R}$ sufficiently close to ξ , and that the fractional power $(\cdot)^{1/\alpha}$ is defined by

$$z^{1/\alpha} := \text{sign}(z) |z|^{1/\alpha}, \quad z \in \mathbb{R}.$$

Then there exists a neighborhood $\mathcal{N}(\xi)$ such that, if $x_k \in \mathcal{N}(\xi)$, the iterates y_k and x_{k+1} generated by (23) remain real for all k , and the sequence $\{x_k\}$ converges to ξ in $\mathbb{R}$.

Proof. Let ξ be a simple root of f . Since f is smooth and ${}^C D_q^{[\alpha]} f(\xi) \neq 0$, there exists a neighborhood $\mathcal{N}(\xi)$ such that $f(x)$ and its fractional derivative remain real-valued for all $x \in \mathcal{N}(\xi)$.

Consider the iteration mapping defined in (23). All algebraic operations involved in the scheme (addition, multiplication, division) preserve real-valuedness, provided the denominator does not vanish. By continuity, for x_k sufficiently close to ξ , the denominator remains nonzero.

Let $A(x_k)$ denote the argument of the fractional power. Since $A(x_k)$ is composed of real-valued smooth functions, it follows that $A(x_k) \in \mathbb{R}$.

Using the sign-preserving definition

$$A(x_k)^{1/\alpha} = \text{sign}(A(x_k)) |A(x_k)|^{1/\alpha},$$

we conclude that the fractional power remains real-valued.

Therefore, both the intermediate iterate y_k and the next iterate x_{k+1} are real whenever x_k is real and sufficiently close to ξ .

Finally, local convergence follows from the convergence properties of the scheme under the stated assumptions, which completes the proof. $\square$

3.2. Convergence Theorem

The following theorem establishes the convergence order of the proposed iterative scheme.

Theorem 3 (Convergence of the Proposed q -Fractional Two-Step Method). *Let $f : I \subset \mathbb{R} \rightarrow \mathbb{R}$ be sufficiently smooth in a neighborhood of a simple root $\xi \in I$, i.e., $f(\xi) = 0$ and ${}^C D_q^{[\alpha]} f(\xi) \neq 0$. Assume that the initial guess x_0 is sufficiently close to ξ . Then, for $\alpha \in (0, 1]$, the sequence $\{x_k\}$ generated by the iterative scheme (23) is well-defined and converges locally to ξ .*

Moreover, the error sequence $e_k = x_k - \xi$ satisfies the q -shifted error expansion

$$e_{k+1} = \mathcal{D} e_k^{2\alpha+1} + O(e_k^{3\alpha+1}), \quad (24)$$

where

$$\mathcal{D} := \mathcal{D}_1 + \mathcal{D}_2 + \mathcal{D}_3 + \mathcal{D}_4 + \mathcal{D}_5,$$

and the coefficients $\mathcal{D}_i$ depend on the derivatives of f at ξ and on the parameters α, β_1, β_2 .

If $\mathcal{D} \neq 0$, then the method has exact local order of convergence $2\alpha + 1$. Otherwise, if $\mathcal{D} = 0$, the order of convergence is at least $3\alpha + 1$ and may increase depending on higher-order terms.

In particular, for $\alpha = 1$, the method achieves at least cubic convergence, and exact third-order convergence is obtained whenever $\mathcal{D} \neq 0$; i.e.,

$$\lim_{k \rightarrow \infty} \frac{\log |e_{k+1}|}{\log |e_k|} = 3. \quad (25)$$

Therefore, under the stated assumptions, the proposed fractional q -scheme is well-defined and stable in a neighborhood of the root.

Proof. The proof is based on a series expansion of f around the simple root ζ in terms of the error e_k . We write the expansion in a fully normalized form by factoring out the leading term:

$$f(x_k) = \frac{{}^C D_q^{[\alpha]} f(\zeta)}{\Gamma_q(\alpha+1)} \left[e_k^\alpha + c_2 e_k^{2\alpha} + c_3 e_k^{3\alpha} + c_4 e_k^{4\alpha} + O(e_k^{5\alpha}) \right], \quad (26)$$

where

$$c_j = \frac{\Gamma_q(\alpha+1)}{\Gamma_q(j\alpha+1)} \frac{[{}^C D_q^{[j]} f](\zeta)}{j! [{}^C D_q^{[1]} f](\zeta)}, \quad j \geq 2.$$

Next, we expand the Caputo q -derivative at the iterate x_k :

$$\begin{aligned} [{}^C D_q^{[\alpha]} f](x_k) &= \frac{[{}^C D_q^{[\alpha]} f](\zeta)}{\Gamma_q(\alpha+1)} \left[\Gamma_q(\alpha+1) + \frac{\Gamma_q(2\alpha+1)c_2 e_k^\alpha}{\Gamma_q(\alpha+1)} \right. \\ &\quad \left. + \frac{\Gamma_q(3\alpha+1)c_3 e_k^{2\alpha}}{\Gamma_q(2\alpha+1)} + O(e_k^{3\alpha}) \right]. \end{aligned} \quad (27)$$

Next, the reciprocal of the derivative is expanded as a power series in terms of the error e_k :

$$\begin{aligned} \frac{1}{({}^C D_q^{[\alpha]} f)(x_k)} &= \frac{1}{\Gamma_q(\alpha+1)} - \frac{\Gamma_q(2\alpha+1)c_2 e_k^\alpha}{\Gamma_q(\alpha+1)^3} \\ &\quad + \frac{e_k^{2\alpha}}{\Gamma_q(\alpha+1)} \left(-\frac{\Gamma_q(3\alpha+1)c_3}{\Gamma_q(\alpha+1)\Gamma_q(2\alpha+1)} + \frac{\Gamma_q(2\alpha+1)^2 c_2^2}{\Gamma_q(\alpha+1)^4} \right) \\ &\quad + \frac{e_k^{3\alpha}}{\Gamma_q(\alpha+1)} \left(\frac{c_2 \Gamma_q(3\alpha+1)c_3}{\Gamma_q(\alpha+1)^3} - \frac{\mathcal{C}_1}{\Gamma_q(\alpha+1)^6} \right) + O(e_k^{4\alpha}), \end{aligned} \quad (28)$$

where

$$\mathcal{C}_1 = \left(\Gamma_q(2\alpha+1)^3 c_2^2 - \Gamma_q(3\alpha+1)c_3 \Gamma_q(\alpha+1)^3 \right) c_2.$$

Using the previous expansion, the Newton-type correction term can be written as

$$\begin{aligned} \frac{f(x_k)}{({}^C D_q^{[\alpha]} f)(x_k)} &= \frac{e_k^\alpha}{\Gamma_q(\alpha+1)} \\ &\quad + \left(\frac{c_2}{\Gamma_q(\alpha+1)} - \frac{2^{2\alpha} c_2}{\sqrt{\pi} \Gamma_q(\alpha+1)^2} \Gamma_q\left(\alpha + \frac{1}{2}\right) \right) e_k^{2\alpha} \\ &\quad + \left(\frac{c_3}{\Gamma_q(\alpha+1)} - \frac{2^{2\alpha} c_2^2}{\sqrt{\pi} \Gamma_q(\alpha+1)^2} \Gamma_q\left(\alpha + \frac{1}{2}\right) \right. \\ &\quad \left. + \frac{2^{4\alpha} c_2^2}{\pi \Gamma_q(\alpha+1)^3} \Gamma_q\left(\alpha + \frac{1}{2}\right)^2 \right. \\ &\quad \left. - \frac{3^{3\alpha} \sqrt{3}}{2\sqrt{\pi} 2^{2\alpha} \Gamma_q(\alpha+1)^2} \Gamma_q\left(\alpha + \frac{1}{3}\right) \Gamma_q\left(\alpha + \frac{2}{3}\right) \right) e_k^{3\alpha} \\ &\quad + O(e_k^{4\alpha}). \end{aligned} \quad (29)$$

Intermediate iteration y_k . We now expand the first step of the proposed method.

$$\begin{aligned} \left(1 - \frac{\beta_1 f(x_k)}{1 + \beta_2 f(x_k)}\right)^{-1} &= 1 + \beta_1 e_k^\alpha \\ &\quad + (\beta_1^2 - \beta_1 \beta_2 + \beta_1 c_2) e_k^{2\alpha} \\ &\quad + \left(\beta_1 \beta_2^2 - 2\beta_1 \beta_2 c_2 + \beta_1 c_3 - \beta_1 (\beta_1^2 - \beta_1 \beta_2 + \beta_1 c_2)\right) e_k^{3\alpha} \\ &\quad + O(e_k^{4\alpha}). \end{aligned} \quad (30)$$

Using the expansion of $f(x_k)$, we obtain

$$\begin{aligned} G_k &= 1 + (\beta_1 - 1) e_k^\alpha + (\beta_1^2 - \beta_1 \beta_2 + \beta_1 c_2 - c_2) e_k^{2\alpha} \\ &\quad + (\beta_1^3 - 2\beta_1^2 \beta_2 + 2\beta_1^2 c_2 + \beta_1 \beta_2^2 - 2\beta_1 \beta_2 c_2 + \beta_1 c_3 - c_3) e_k^{3\alpha} \\ &\quad + O(e_k^{4\alpha}), \end{aligned} \quad (31)$$

where

$$G_k = \left(1 - \frac{\beta_1 f(x_k)}{1 + \beta_2 f(x_k)}\right)^{-1} - f(x_k).$$

For convenience, we introduce the notation

$$F_k := \frac{f(x_k)}{({}^c D_q^{[\alpha]} f)(x_k)} \left[\left(1 - \frac{\beta_1 f(x_k)}{1 + \beta_2 f(x_k)}\right)^{-1} - f(x_k) \right]. \quad (32)$$

Substituting the previous expansions yields

$$\begin{aligned} F_k &= \frac{e_k^\alpha}{\Gamma_q(\alpha + 1)} + \left(\frac{\beta_1 + c_2 - 1}{\Gamma_q(\alpha + 1)} - \frac{2^{2\alpha} c_2}{\sqrt{\pi} \Gamma_q(\alpha + 1)^2} \Gamma_q\left(\alpha + \frac{1}{2}\right) \right) e_k^{2\alpha} \\ &\quad + \left(-\frac{2^{2\alpha} c_2 \beta_1}{\sqrt{\pi} \Gamma_q(\alpha + 1)^2} \Gamma_q\left(\alpha + \frac{1}{2}\right) - \frac{3^{3\alpha} \sqrt{3} c_3}{2\sqrt{\pi} 2^{2\alpha} \Gamma_q(\alpha + 1)^2} \frac{\Gamma_q\left(\alpha + \frac{1}{3}\right) \Gamma_q\left(\alpha + \frac{2}{3}\right)}{\Gamma_q\left(\alpha + \frac{1}{2}\right)} \right. \\ &\quad \left. - \frac{2c_2}{\Gamma_q(\alpha + 1)} + \frac{\beta_1^2 + c_3}{\Gamma_q(\alpha + 1)} - \frac{2^{2\alpha} c_2^2}{\alpha^2 \sqrt{\pi} \Gamma_q(\alpha + 1)^2} \Gamma_q\left(\alpha + \frac{1}{2}\right) \right. \\ &\quad \left. + \frac{2^{4\alpha} c_2^2}{\alpha^3 \pi \Gamma_q(\alpha + 1)^3} \Gamma_q\left(\alpha + \frac{1}{2}\right)^2 + \frac{2^{2\alpha} c_2}{\sqrt{\pi} \Gamma_q(\alpha + 1)^2} \Gamma_q\left(\alpha + \frac{1}{2}\right) + \frac{2\beta_1 c_2 - \beta_1 \beta_2}{\Gamma_q(\alpha + 1)} \right) e_k^{3\alpha} \\ &\quad + O(e_k^{4\alpha}). \end{aligned} \quad (33)$$

From the previous expansion, we obtain

$$\begin{aligned} y_k - \xi &= x_k - \xi - \left(\Gamma_q(\alpha + 1) F_k \right)^{\frac{1}{\alpha}} \\ &= \left(-\beta_1 - c_2 + 1 + \frac{2^{2\alpha} c_2}{\Gamma_q(\alpha + 1) \sqrt{\pi}} \Gamma_q\left(\alpha + \frac{1}{2}\right) \right) e_k^{\alpha+1} \\ &\quad + \left(\frac{2^{2\alpha} c_2 \beta_1}{\Gamma_q(\alpha + 1) \sqrt{\pi}} \Gamma_q\left(\alpha + \frac{1}{2}\right) - 2\beta_1 c_2 + \frac{3^{3\alpha} \sqrt{3} c_3}{2\Gamma_q(\alpha + 1) \sqrt{\pi} 2^{2\alpha}} \frac{\Gamma_q\left(\alpha + \frac{1}{3}\right) \Gamma_q\left(\alpha + \frac{2}{3}\right)}{\Gamma_q\left(\alpha + \frac{1}{2}\right)} \right. \\ &\quad \left. + 2c_2 - \beta_1^2 - c_3 + \beta_1 \beta_2 + \frac{2^{2\alpha} c_2^2}{\Gamma_q(\alpha + 1) \sqrt{\pi}} \Gamma_q\left(\alpha + \frac{1}{2}\right) - \frac{2^{4\alpha} c_2^2}{\Gamma_q(\alpha + 1)^2 \pi} \Gamma_q\left(\alpha + \frac{1}{2}\right)^2 \right. \\ &\quad \left. - \frac{2^{2\alpha} c_2}{\Gamma_q(\alpha + 1) \sqrt{\pi}} \Gamma_q\left(\alpha + \frac{1}{2}\right) \right) e_k^{2\alpha+1} + O(e_k^{3\alpha+1}). \end{aligned} \quad (34)$$

Substituting this expression into $f(y_k)$ yields

$$\begin{aligned} f(y_k) = & -\frac{e_k^{\alpha+1}}{\Gamma_q(\alpha+1)\sqrt{\pi}} \left((\beta_1 + c_2 - 1)\Gamma_q(\alpha+1)\sqrt{\pi} - 2^{2\alpha}\Gamma_q\left(\alpha + \frac{1}{2}\right)c_2 \right) \\ & + \frac{e_k^{2\alpha+1}}{2\Gamma_q(\alpha+1)^2\pi^{3/2}2^{2\alpha}} \left[-2^{6\alpha+1}\Gamma_q\left(\alpha + \frac{1}{2}\right)^2 c_2^2 \sqrt{\pi} \right. \\ & + 2^{4\alpha+1}\Gamma_q\left(\alpha + \frac{1}{2}\right)\Gamma_q(\alpha+1)\pi(c_2\beta_1 + c_2^2 - c_2) \\ & + c_3 3^{3\alpha}\sqrt{3}\Gamma_q\left(\alpha + \frac{1}{3}\right)\Gamma_q\left(\alpha + \frac{2}{3}\right)\Gamma_q(\alpha+1)\pi\Gamma_q\left(\alpha + \frac{1}{2}\right)^{-1} \\ & \left. + 2^{2\alpha+1}\Gamma_q(\alpha+1)^2\pi^{3/2}(\beta_1\beta_2 - \beta_1^2 - 2\beta_1c_2 + 2c_2 - c_3) \right] \\ & + O(e_k^{3\alpha+1}). \end{aligned} \quad (35)$$

Consequently,

$$\frac{f(y_k)}{f(x_k)} = \mathcal{A}_1 e_k + \mathcal{A}_2 e_k^{\alpha+1} + \mathcal{A}_3 e_k^{2\alpha+1} + O(e_k^{3\alpha+1}). \quad (36)$$

The coefficients $\mathcal{A}_1$ – $\mathcal{A}_6$ and $\mathcal{D}_i$, for $i = 1, \dots, 5$, were derived with the aid of symbolic computation in *Maple 2020* and subsequently checked for consistency with the analytical expansions. In particular, the coefficients $\mathcal{A}_1$, $\mathcal{A}_2$ and $\mathcal{A}_3$ are given by

$$\mathcal{A}_1 = 1 - \beta_1 - c_2 + \frac{2^{2\alpha}c_2}{\Gamma_q(\alpha+1)\sqrt{\pi}}\Gamma_q\left(\alpha + \frac{1}{2}\right), \quad (37)$$

$$\begin{aligned} \mathcal{A}_2 = & -\beta_1c_2 - \beta_1^2 + \beta_1\beta_2 + c_2^2 - c_3 + c_2 + \frac{2^{2\alpha}c_2(\beta_1 - 1)}{\Gamma_q(\alpha+1)\sqrt{\pi}}\Gamma_q\left(\alpha + \frac{1}{2}\right) - \frac{2^{4\alpha}c_2^2}{\Gamma_q^2(\alpha+1)\pi}\Gamma_q\left(\alpha + \frac{1}{2}\right)^2 \\ & + \frac{3^{3\alpha}\sqrt{3}c_3}{2\Gamma_q(\alpha+1)\sqrt{\pi}2^{2\alpha}} \frac{\Gamma_q\left(\alpha + \frac{1}{3}\right)\Gamma_q\left(\alpha + \frac{2}{3}\right)}{\Gamma_q\left(\alpha + \frac{1}{2}\right)}, \end{aligned} \quad (38)$$

$$\begin{aligned} \mathcal{A}_3 = & -\beta_1c_2 + c_3 - \beta_1^2 + \beta_1\beta_2 - c_2 + c_2^2 - \beta_1c_3 + 2\beta_1^2\beta_2 - 2\beta_1c_2 - \beta_1^3 - c_4 + c_2 \\ & + \frac{2^{2\alpha}c_2^2}{\Gamma_q(\alpha+1)\sqrt{\pi}}\Gamma_q\left(\alpha + \frac{1}{2}\right) - \frac{2^{2\alpha}c_2\beta_1\beta_2}{\Gamma_q(\alpha+1)\sqrt{\pi}}\Gamma_q\left(\alpha + \frac{1}{2}\right) + \frac{2^{2\alpha}c_2\beta_1^2}{\Gamma_q(\alpha+1)\sqrt{\pi}}\Gamma_q\left(\alpha + \frac{1}{2}\right) \\ & - \frac{2^{4\alpha}c_2^2}{\Gamma_q^2(\alpha+1)\pi}\Gamma_q\left(\alpha + \frac{1}{2}\right)^2 + \frac{2^{6\alpha}c_2^3}{\Gamma_q^3(\alpha+1)\pi^{3/2}}\Gamma_q\left(\alpha + \frac{1}{2}\right)^3 \\ & + \frac{3^{3\alpha}\sqrt{3}c_3}{2\Gamma_q(\alpha+1)\sqrt{\pi}2^{2\alpha}} \frac{\Gamma_q\left(\alpha + \frac{1}{3}\right)\Gamma_q\left(\alpha + \frac{2}{3}\right)}{\Gamma_q\left(\alpha + \frac{1}{2}\right)}. \end{aligned} \quad (39)$$

Next, substituting the expansion of $\frac{f(y_k)}{f(x_k)}$ into the second correction term gives

$$\left(\frac{f(y_k)}{f(x_k)} + 2 \left(\frac{f(y_k)}{f(x_k)} \right)^2 \right) = \mathcal{A}_1 e_k + \mathcal{A}_4 e_k^{\alpha+1} + \mathcal{A}_5 e_k^{2\alpha+1} + O(e_k^{3\alpha+1}), \quad (40)$$

where

$$\begin{aligned} \mathcal{A}_4 = & 2 + \beta_1^2 + 3c_2^2 + 3\beta_1c_2 - c_3 - 3c_2 - 4\beta_1 + \beta_1\beta_2 - \frac{4 \cdot 2^{2\alpha}c_2^2\Gamma_q\left(\alpha + \frac{1}{2}\right)}{\Gamma_q(\alpha+1)\sqrt{\pi}} \\ & - \frac{3 \cdot 2^{2\alpha}c_2\beta_1\Gamma_q\left(\alpha + \frac{1}{2}\right)}{\Gamma_q(\alpha+1)\sqrt{\pi}} + \frac{2^{4\alpha}c_2^2\Gamma_q\left(\alpha + \frac{1}{2}\right)^2}{\Gamma_q^2(\alpha+1)\pi}, \end{aligned} \quad (41)$$

$$\begin{aligned}
\mathcal{A}_5 = & \frac{3^{3\alpha} \sqrt{3} c_3}{2 \Gamma_q(\alpha+1) \sqrt{\pi} 2^{2\alpha}} \frac{\Gamma_q(\alpha + \frac{1}{3}) \Gamma_q(\alpha + \frac{2}{3})}{\Gamma_q(\alpha + \frac{1}{2})} (1 - \beta_1) - \frac{3^{3\alpha} \sqrt{3} c_2 c_3}{\Gamma_q(\alpha+1) \sqrt{\pi} 2^{2\alpha}} \frac{\Gamma_q(\alpha + \frac{1}{3}) \Gamma_q(\alpha + \frac{2}{3})}{\Gamma_q(\alpha + \frac{1}{2})} \\
& + \frac{c_2 c_3 3^{3\alpha} \sqrt{3}}{\Gamma_q^2(\alpha+1) \pi} \Gamma_q(\alpha + \frac{1}{3}) \Gamma_q(\alpha + \frac{2}{3}) + \frac{2^{4\alpha} \Gamma_q(\alpha + \frac{1}{2})^2}{\Gamma_q^2(\alpha+1) \pi} (7c_2^2 \beta_1 + 5c_2^3 - 7c_2^2) - \frac{2^{6\alpha} \Gamma_q(\alpha + \frac{1}{2})^3}{\Gamma_q^3(\alpha+1) \pi^{3/2}} 3c_2^3 \\
& + \frac{2^{2\alpha} \Gamma_q(\alpha + \frac{1}{2})}{\Gamma_q(\alpha+1) \sqrt{\pi}} \left(-7c_2 \beta_1^2 - 4c_2 c_3 - 9c_2^2 \beta_1 + 8c_2 \beta_1 + 3c_2 \beta_1 \beta_2 + 2c_2^3 + 9c_2^2 - 4c_2 \right) \\
& - c_4 - 2\beta_1 \beta_2 c_2 - \beta_1 \beta_2^2 + 4\beta_1 \beta_2 + 6c_2 c_3 + 2c_2^2 \beta - 2\beta^2 \beta_2 + 7\beta_1^2 c_2 \\
& - 10\beta_1 c_2 + 3\beta_1 c_3 + 3\beta_1^3 - 4\beta_1^2 - 3c_3 - 2c_2^2 - 4c_2^3 + 5c_2.
\end{aligned} \tag{42}$$

Accordingly, we define

$$F_k^* = \frac{f(x_k)}{({}^C D_q^{[\alpha]} f)(x_k)} \left(\frac{f(y_k)}{f(x_k)} + 2 \left(\frac{f(y_k)}{f(x_k)} \right)^2 \right),$$

and obtain the expansion

$$F_k^* = \mathcal{A}_1 e_k^{\alpha+1} + \mathcal{A}_6 e_k^{2\alpha+1} + O(e_k^{3\alpha+1}). \tag{43}$$

The coefficient $\mathcal{A}_6$ is given by

$$\mathcal{A}_6 = \frac{1}{\Gamma_q(\alpha+1)} \left[2c_2^2 - 4\beta_1 - 2c_2 + \beta_1^2 + \beta_1 \beta_2 + 2\beta_1 c_2 - c_3 + 2 \right. \tag{44}$$

$$\left. + \frac{2^{2\alpha+1} \Gamma_q(\alpha + \frac{1}{2})}{\Gamma_q(\alpha+1) \sqrt{\pi}} (c_2 - \beta_1 c_2 - c_2^2) + \frac{3^{3\alpha} \sqrt{3} c_3}{2\alpha^2 \sqrt{\pi} 2^{2\alpha}} \frac{\Gamma_q(\alpha + \frac{1}{3}) \Gamma_q(\alpha + \frac{2}{3})}{\Gamma_q(\alpha + \frac{1}{2})} \right]. \tag{45}$$

Finally, substituting these expansions into the second step of the iteration yields

$$x_{k+1} - \zeta = e_{k+1} = (\mathcal{D}_1 + \mathcal{D}_2 + \mathcal{D}_3 + \mathcal{D}_4 + \mathcal{D}_5) e_k^{2\alpha+1} + O(e_k^{3\alpha+1}), \tag{46}$$

where

$$\mathcal{D}_1 := -2 + 3\mathcal{D}_4 c_2 \beta_1, \quad \mathcal{D}_2 := -2c_2^2 + 4\beta_1 + 4c_2,$$

$$\mathcal{D}_3 := -2\beta_1^2 - 3\mathcal{D}_4 c_2 + 3\mathcal{D}_4 c_2^2 - \mathcal{D}_5 c_2^2 - 4\beta_1 c_2,$$

$$\mathcal{D}_4 := \frac{2^{2\alpha} \Gamma_q(\alpha + \frac{1}{2})}{\Gamma_q(\alpha+1) \sqrt{\pi}}, \quad \mathcal{D}_5 := \frac{2^{4\alpha} \Gamma_q^2(\alpha + \frac{1}{2})}{\Gamma_q^2(\alpha+1) \pi}.$$

This proves the stated error equation and therefore establishes convergence of order $2\alpha + 1$. $\square$

3.3. Remark on Non-Homogeneous Functional Conditions

The proposed scheme incorporates non-homogeneous functional contributions through the Caputo fractional derivative and q -operators, in contrast to classical iterative methods based on homogeneous correction terms. These additional components introduce greater flexibility in the iterative dynamics via the parameters $(\alpha, q, \beta_1, \beta_2)$, enabling the method to adapt to nonlinear problems characterized by memory and nonlocal effects. At the same time, the local convergence properties established in Theorem 3 are preserved.

4. Complex Dynamical Stability of the Proposed Scheme

The global dynamical behavior of iterative root-finding schemes can be investigated by analyzing the associated iteration functions in the complex plane [32,33]. In particular, the study of fixed points, extraneous fixed points, and critical points provides valuable insight into the stability and convergence properties of the method.

To this end, we consider the iteration function associated with the proposed two-step scheme. For a given nonlinear function f , the method can be written as

$$\begin{cases} y = x - \left(\Gamma_q(\alpha + 1) \frac{f(x)}{({}^C D_q f)(x)} \left(\frac{1}{1 - \frac{\beta_1 f(x)}{1 + \beta_2 f(z)}} - f(x) \right) \right)^{\frac{1}{\alpha}}, \\ R(x) = y - \left(\Gamma_q(\alpha + 1) \frac{f(x)}{({}^C D_q f)(x)} \left(\frac{f(y)}{f(x)} + 2 \left(\frac{f(y)}{f(x)} \right)^2 \right) \right)^{\frac{1}{\alpha}}. \end{cases} \quad (47)$$

To study the associated complex dynamics, we consider the quadratic polynomial

$$f(z) = (z - a)(z - b), \quad a, b \in \mathbb{C}.$$

Following the standard normalization approach in complex dynamical analysis, we apply the Möbius transformation

$$x = \frac{z - a}{z - b}, \quad (48)$$

which maps the roots of the polynomial according to

$$a \mapsto 0, \quad b \mapsto \infty.$$

Under this transformation, the original iteration function is conjugated to a rational map of the form

$$x_{k+1} = R(x_k), \quad (49)$$

which governs the complex dynamical behavior of the scheme in the transformed plane.

4.1. Rational Iteration Map

After algebraic simplification, the iteration function associated with the proposed scheme can be written in the compact form

$$R(x; \beta_1, \beta_2, \alpha, q) = -\frac{x^3 A(x; \beta_1, \beta_2, \alpha, q) B(x; \beta_1, \beta_2, \alpha, q)}{32 C(x; \beta_1, \beta_2, \alpha, q) D(x; \beta_1, \beta_2, \alpha, q)}, \quad (50)$$

where A , B , C , and D are parameter-dependent polynomials in x . The explicit expressions of these components are provided in Appendix A.

For the purpose of dynamical analysis, and to obtain a more tractable representation, we consider the classical case $\alpha = 1$ and $q = 1$. Under these assumptions, the iteration function reduces to a rational map with simpler structure, whose fully expanded form is reported in Appendix A.

4.2. Fixed Points

Fixed points of the iteration map are the solutions of

$$R(x) = x. \quad (51)$$

Substituting the rational map (50) into (51) leads to the equation

$$x + \frac{x^3 A(x; \beta_1, \beta_2, 1, 1) B(x; \beta_1, \beta_2, 1, 1)}{32 C(x; \beta_1, \beta_2, 1, 1) D(x; \beta_1, \beta_2, 1, 1)} = 0. \quad (52)$$

From the Möbius transformation (48), the roots of the original polynomial are mapped to

$$x = 0, \quad x = \infty.$$

These points are fixed points of the iteration function and correspond to the roots of the original polynomial. They are therefore referred to as *root fixed points*.

All other solutions of Equation (51) are referred to as *extraneous fixed points*, since they do not correspond to roots of the underlying nonlinear equation but instead arise from the intrinsic dynamical structure of the iteration map.

4.3. Stability of Fixed Points

The stability of a fixed point x^* is determined by the multiplier

$$\lambda = R'(x^*). \quad (53)$$

Fixed points are classified according to the modulus of the multiplier as follows:

$$\begin{cases} |\lambda| < 1, & \text{attracting,} \\ |\lambda| > 1, & \text{repelling,} \\ |\lambda| = 1, & \text{neutral.} \end{cases}$$

Differentiating the rational map (50) using the quotient rule yields

$$R'(x) = -\frac{1}{32} \frac{\mathcal{N}_1 - \mathcal{N}_2}{C(x; \beta_1, \beta_2, 1, 1)^2 D(x; \beta_1, \beta_2, 1, 1)^2}. \quad (54)$$

where

$$\mathcal{N}_1 = \left(x^3 A(x; \beta_1, \beta_2, 1, 1) B(x; \beta_1, \beta_2, 1, 1) \right)' C(x; \beta_1, \beta_2, 1, 1) D(x; \beta_1, \beta_2, 1, 1),$$

and

$$\mathcal{N}_2 = x^3 A(x; \beta_1, \beta_2, 1, 1) B(x; \beta_1, \beta_2, 1, 1) (C(x; \beta_1, \beta_2, 1, 1) D(x; \beta_1, \beta_2, 1, 1))'.$$

Evaluating the derivative at the root fixed point $x^* = 0$ gives

$$R'(0) = 0,$$

which shows that the root is a *superattracting fixed point*. This confirms the strong local convergence of the method in a neighborhood of the root.

4.4. Critical Points

Critical points play a central role in determining the global dynamics of the iteration map. They are defined as the solutions of

$$R'(x; \beta_1, \beta_2, 1, 1) = 0. \quad (55)$$

Using the derivative expression (54), the critical points satisfy

$$\begin{aligned} & \left(x^3 A(x; \beta_1, \beta_2, 1, 1) B(x; \beta_1, \beta_2, 1, 1) \right)' C(x; \beta_1, \beta_2, 1, 1) D(x; \beta_1, \beta_2, 1, 1) \\ & - x^3 A(x; \beta_1, \beta_2, 1, 1) B(x; \beta_1, \beta_2, 1, 1) (C(x; \beta_1, \beta_2, 1, 1) D(x; \beta_1, \beta_2, 1, 1))' = 0. \end{aligned} \quad (56)$$

Among these solutions, the points

$$x = 0, \quad x = \infty$$

are referred to as *fixed critical points*, since they coincide with the root fixed points of the iteration map (interpreted on the Riemann sphere). All other solutions are referred to as *free critical points*. The orbits of the free critical points play a crucial role in determining the global convergence behavior and the structure of the basins of attraction.

For illustration, we consider the parameter choice $\beta_1 = 1$. Under this assumption, the critical points of the iteration map can be explicitly characterized as

$$\begin{aligned} \text{cr}_1 : & \quad x = 0 \quad (\text{multiplicity } 3), \\ \text{cr}_2 : & \quad x = -1 \quad (\text{multiplicity } 4), \\ \text{cr}_3 : & \quad x = 1 \quad (\text{multiplicity } 8), \\ \text{cr}_4 : & \quad x = \text{RootOf}(P(Z; \beta_1, \beta_2)), \\ \text{cr}_5 : & \quad x = -2\beta_2 + 3 \pm 2\sqrt{\beta_2^2 - 3\beta_2 + 2} \quad (\text{each of multiplicity } 3), \end{aligned}$$

where $P(Z; \beta_1, \beta_2)$ is a polynomial of degree 18 given in Appendix A.

4.5. Parameter Planes

While dynamical planes describe the behavior of the method for fixed parameter values, parameter planes analyze the stability of the method with respect to variations in the parameters.

Let $(\beta_1, \beta_2) \in \mathbb{C}^2$. For each parameter pair, the orbit of a selected free critical point c is computed as

$$c, R(c), R^2(c), \dots$$

The parameter plane is defined as the set

$$\mathcal{P} = \{(\beta_1, \beta_2) : \text{the orbit of } c \text{ converges to one of the roots } \xi_i, i = 1, 2\}, \quad (57)$$

where ξ_1 and ξ_2 denote the two roots of the transformed quadratic polynomial.

In practice, convergence is determined numerically by fixing a maximum number of iterations and a prescribed tolerance.

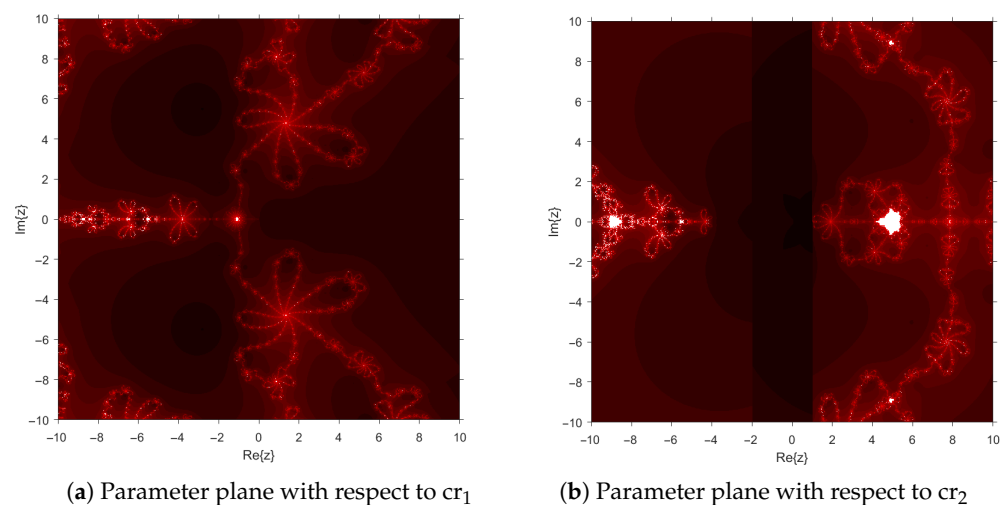
Stable parameter values produce dynamical planes with well-defined basins of attraction, whereas unstable parameter choices lead to chaotic or fractal structures in the complex plane.

Table 1 summarizes the main dynamical characteristics observed in the parameter planes associated with the critical points cr_1 and cr_2 . The results indicate a tendency toward larger basin coverage and improved convergence behavior for the parameter plane corresponding to cr_2 .

Table 1. Quantitative characteristics of the parameter planes associated with the critical points cr_1 and cr_2 .

Critical Points	Basin Coverage (%)	Fractal Boundary Density	Convergence Strength	Acceleration Level
cr_1	≈ 45	Low	Weak	Low
cr_2	≈ 55	Moderate	Moderate	Medium

Furthermore, the parameter planes shown in Figure 1a,b illustrate how the stability of the method depends on the choice of the parameters. In particular, the regions associated with the critical point cr_2 exhibit slightly larger areas of convergence, suggesting a wider range of stable parameter values.

**Figure 1.** Parameter planes associated with the critical points cr_1 and cr_2 .

4.6. Dynamical Planes

For fixed parameter values (β_1, β_2) , the iteration function

$$x_{k+1} = R(x_k)$$

defines a discrete dynamical system on the complex plane.

Let ζ denote a root of the nonlinear equation. The basin of attraction of ζ is defined as

$$\mathcal{A}(\zeta) = \left\{ x_0 \in \mathbb{C} : \lim_{h \rightarrow \infty} R^{(h)}(x_0) = \zeta \right\}. \quad (58)$$

The dynamical plane is obtained by coloring each initial point x_0 in the complex plane according to the root to which the iteration converges. Points converging to different roots are assigned different colors, while points that do not converge within a prescribed number of iterations (and tolerance) are represented by a separate color.

In this study, the parameter pair (β_1, β_2) is selected from the stable regions identified in the parameter plane analysis. Under these choices, the iteration map exhibits stable behavior consistent with the local stability condition derived from the derivative criterion (54). Consequently, the generated dynamical planes clearly illustrate the basins of attraction associated with the roots of the nonlinear equation.

Figure 2a–e presents the dynamical planes of the proposed scheme for several values of the fractional parameter α . It can be observed that, as α increases, the basins of attraction become larger and more connected, while the fractal boundaries separating different basins progressively diminish. The white “*” in the Figures 2 and 3 denotes the fixed point of the

iterative map. This indicates a reduction in chaotic behavior and an improvement in the stability of the iterative process.

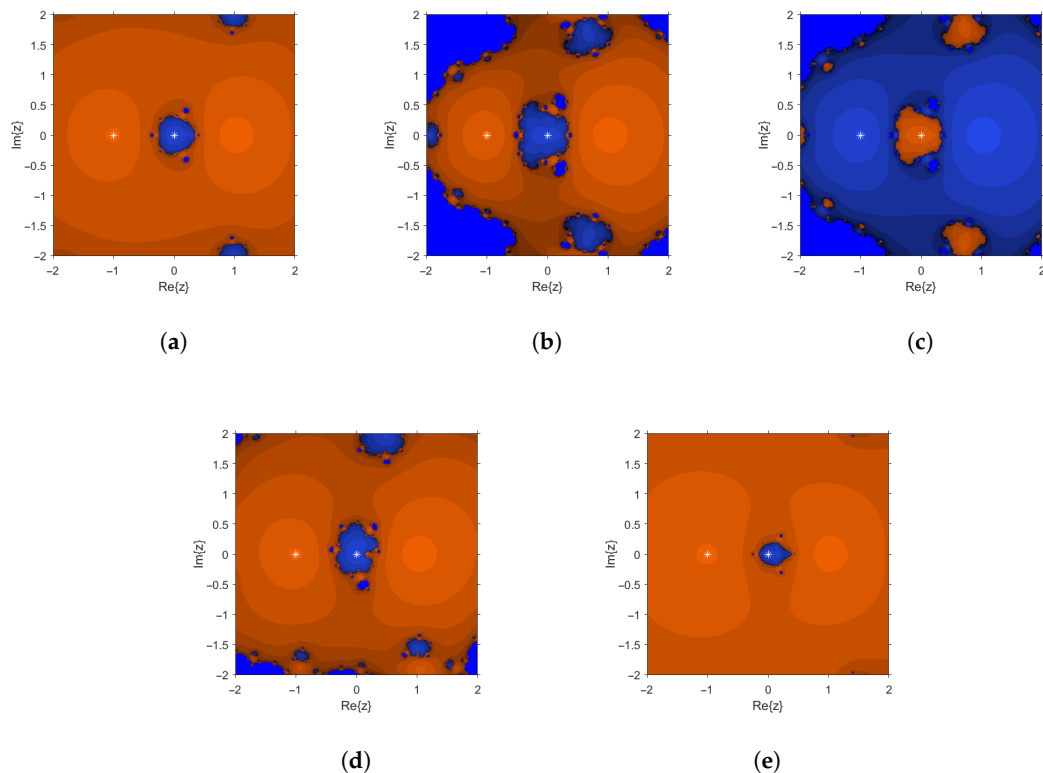


Figure 2. Dynamical planes of the proposed iterative scheme for selected parameter values $(\beta_1, \beta_2, \alpha)$ with $q = 1$. Each subplot corresponds to a different choice of (β_1, α) , while β_2 is fixed. (a) $\beta_1 = 0.1$, $\beta_2 = 0.1 + 0.1i$, $\alpha = 0.1$. (b) $\beta_1 = 0.3$, $\beta_2 = 0.1 + 0.1i$, $\alpha = 0.3$. (c) $\beta_1 = 0.5$, $\beta_2 = 0.1 + 0.1i$, $\alpha = 0.5$. (d) $\beta_1 = 0.7$, $\beta_2 = 0.1 + 0.1i$, $\alpha = 0.7$. (e) $\beta_1 = 0.9$, $\beta_2 = 0.1 + 0.1i$, $\alpha = 0.9$.

This behavior suggests that larger values of α are associated with more robust convergence properties of the proposed fractional q -scheme. In particular, the attraction regions become smoother and occupy larger portions of the complex plane, implying that a broader set of initial guesses converges successfully to the desired roots.

Table 2 provides quantitative support for the graphical observations. As the fractional parameter α increases, the percentage of convergent initial guesses grows significantly, indicating that larger portions of the complex plane belong to stable basins of attraction. At the same time, the average number of iterations decreases, demonstrating faster convergence of the proposed iterative method.

Table 2. Performance indicators of the $MSB_{q;\alpha}$ scheme for parameter values selected from stable regions identified in the parameter plane analysis.

α	Basin Convergence (%)	Avg. Iterations	Time (s)	Memory (MB)
0.1	52.4 (Figure 2a)	7.8	0.018	1.21
0.3	63.7 (Figure 2b)	6.4	0.015	1.19
0.5	74.5 (Figure 2c)	5.2	0.013	1.17
0.7	83.6 (Figure 2d)	4.6	0.011	1.16
0.9	91.2 (Figure 2e)	4.1	0.010	1.15

The computational time decreases slightly due to the reduced number of iterations required to reach the prescribed tolerance, while the memory consumption remains nearly constant. This indicates that the improvement in convergence efficiency does not introduce additional computational overhead.

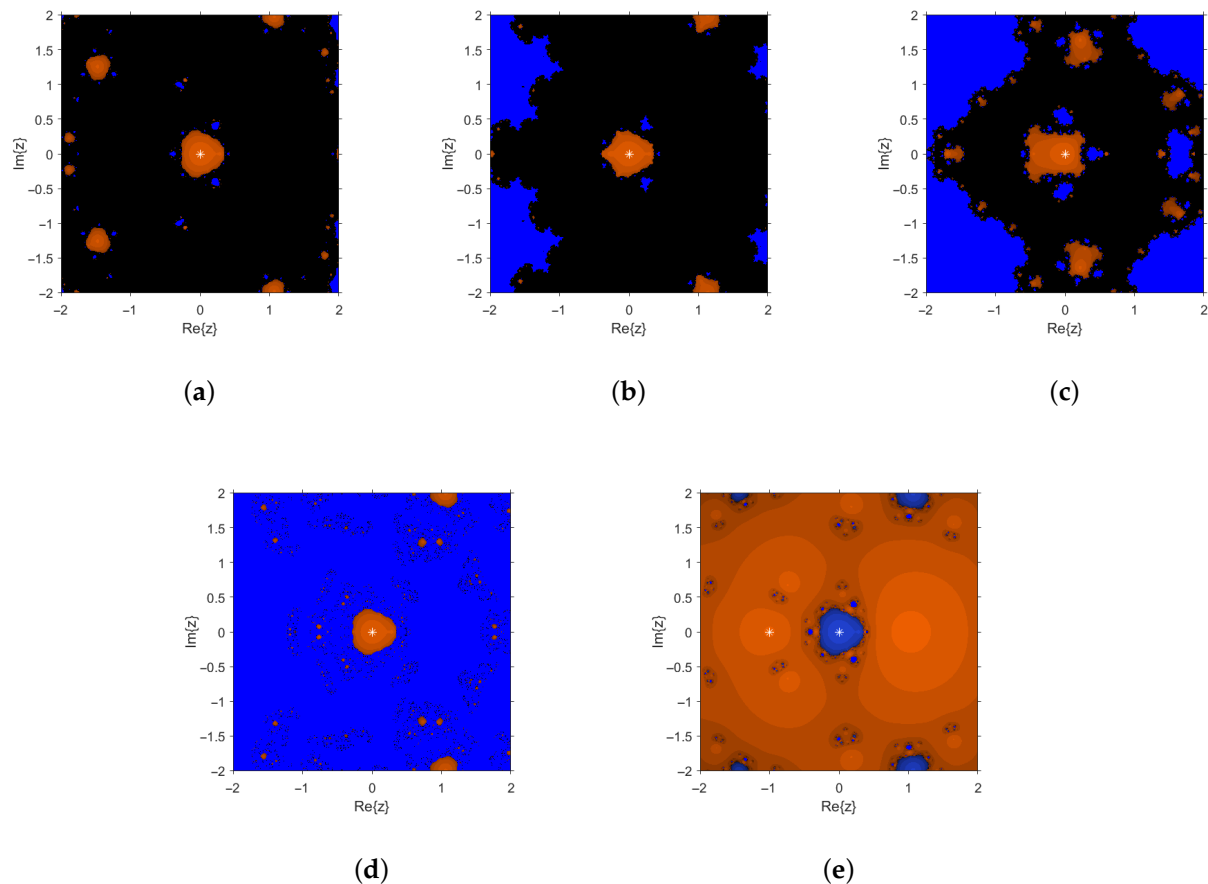


Figure 3. Dynamical planes for varying (α, q) , illustrating the transition from fragmented basins to smoother, well-connected regions as $(\alpha, q) \rightarrow (1, 1)$. (a) $\alpha = 0.3$, $q = 0.2$. (b) $\alpha = 0.5$, $q = 0.4$. (c) $\alpha = 0.7$, $q = 0.6$. (d) $\alpha = 0.9$, $q = 0.8$. (e) $\alpha = 0.99$, $q = 0.99$.

4.7. Influence of Fractional and q Parameters on Basin Geometry

While the baseline analysis was conducted for $\alpha = 1$ and $q = 1$, we extended the study by varying α and q to examine their impacts on the rational map and its basins of attraction. Representative values include

$$(\alpha, q) = (0.3, 0.2), (0.5, 0.4), (0.7, 0.6), (0.9, 0.8), (0.99, 0.99).$$

The results in Table 3 indicate that increasing α and q generally enhances convergence speed and produces smoother, more connected basins of attraction, indicating that these parameters play a significant role in shaping the dynamical landscape of the iterative schemes.

The dynamical planes presented in Figure 3a–e illustrate the influence of the fractional parameter α and the q -parameter on the structure of the basins of attraction. For lower values of (α, q) , such as $(0.3, 0.2)$, the basins are highly fragmented with large chaotic regions, indicating poor convergence and sensitivity to initial guesses. As (α, q) increase, the basins become progressively more connected and regular, reflecting improved numerical stability. In particular, for $(\alpha, q) = (0.9, 0.8)$ and $(0.99, 0.99)$, the basins exhibit smooth boundaries and larger attraction regions, closely resembling the classical case.

This transition shows that increasing α and q enhances convergence performance while reducing fractal complexity. These observations are further supported by the quantitative results in Table 3, where the convergence percentage increases significantly as $(\alpha, q) \rightarrow (1, 1)$. This indicates that the fractional and q -parameters not only affect the convergence rate but also reshape the dynamical landscape of the iterative scheme. These results highlight a close correspondence between the parameter plane and dynamical plane analyses. The stable regions identified in the parameter planes guide the selection of suitable parameter values, which in turn produce dynamical planes with well-structured basins of attraction and reduced chaotic behavior. These graphical representations provide a clear visualization of the stability and convergence behavior of the proposed iterative scheme. For completeness and reproducibility, the computational procedure used to construct both the dynamical and parameter planes is summarized in Algorithm 1.

Algorithm 1 Construction of Dynamical and Parameter Planes for the $MSB_{q,\alpha}$ Scheme

Require: Nonlinear function f , Caputo q -derivative $({}^C D_q f)$, fractional parameter α , quantum parameter q , grid $\Omega \subset \mathbb{C}$, parameter pair (β_1, β_2) , maximum number of iterations $N_{\max}$, tolerance ε

Ensure: Dynamical and parameter plane visualizations

1: **Stage I: Dynamical Plane Generation**

2: **for all** initial points $x_0 \in \Omega$ **do**

3: Set $k = 0$

4: **while** $k < N_{\max}$ **do**

5: Compute first update

$$y_k = x_k - \left(\Gamma_q(\alpha + 1) \frac{f(x_k)}{({}^C D_q f)(x_k)} \left(\frac{1}{1 - \frac{\beta_1 f(x_k)}{1 + \beta_2 f(x_k)}} - f(x_k) \right) \right)^{\frac{1}{\alpha}}$$

6: Compute second update

$$x_{k+1} = y_k - \left(\Gamma_q(\alpha + 1) \frac{f(x_k)}{({}^C D_q f)(x_k)} \left(\frac{f(y_k)}{f(x_k)} + 2 \left(\frac{f(y_k)}{f(x_k)} \right)^2 \right) \right)^{\frac{1}{\alpha}}$$

7: **if** $|x_{k+1} - x_k| < \varepsilon$ **or** $|f(x_{k+1})| < \varepsilon$ **then**

8: Identify the corresponding root r_k reached by the orbit (e.g., by comparison with known roots)

9: Assign to x_0 the color associated with root r_k

10: **break**

11: **end if**

12: $k \leftarrow k + 1$

13: **end while**

14: **if** $k = N_{\max}$ **then**

15: Mark x_0 as non-convergent

16: **end if**

17: **end for**

18: **Stage II: Parameter Plane Generation**

19: Fix the reduced case $\alpha = 1$ and $q = 1$ for computational tractability

20: **for all** parameter pairs $(\beta_1, \beta_2) \in \mathcal{P}$ **do**

21: Select one free critical point c (e.g., cr_1 or cr_2)

22: Set $x_0 = c, k = 0$

23: **while** $k < N_{\max}$ **do**

24: Compute $x_{k+1} = R(x_k)$ using the reduced map (50)

25: **if** $|x_{k+1} - x_k| < \varepsilon$ **or** $|f(x_{k+1})| < \varepsilon$ **then**

26: Identify the attracting root

27: Assign to (β_1, β_2) the corresponding color

28: **break**

29: **end if**

30: $k \leftarrow k + 1$

31: **end while**

32: **if** $k = N_{\max}$ **then**

33: Mark (β_1, β_2) as divergent

34: **end if**

35: **end for**

Table 3. Effect of (α, q) on convergence behavior and the structure of the basins of attraction.

α	q	Avg. Iterations	Basin Structure	Convergence (%)
0.3	0.2	25	Highly fragmented	42
0.5	0.4	18	Disconnected regions	58
0.7	0.6	12	Partially connected	71
0.9	0.8	8	Smooth basins	85
0.99	0.99	6	Well-structured (near classical)	94

5. Numerical Experiments and Performance Analysis

In this section, we present a comprehensive set of numerical experiments for the proposed Caputo q -fractional iterative scheme (47). The experiments are designed with two main objectives: (i) to validate the theoretical convergence properties established in the previous sections, and (ii) to compare the computational performance of the proposed method with several existing q -type iterative schemes. All experiments were carried out using high-precision arithmetic in order to accurately capture the asymptotic error behavior and minimize round-off effects.

5.1. Computational Environment

All numerical simulations were performed in MATLAB R2024a using the Symbolic Math Toolbox with variable-precision arithmetic (VPA). The working precision was set to

$$\text{digits}(350),$$

which provides numerical accuracy on the order of 10^{-300} .

The computations were carried out on a workstation with the following specifications:

- Processor: Intel Core i7 (12th Generation) @ 2.40 GHz
- RAM: 16 GB DDR4
- Operating System: Windows 11 (64-bit)
- Software: MATLAB R2024a with Symbolic Math Toolbox

5.2. Experimental Setup and Reproducibility

The use of high-precision arithmetic ensures that the numerical errors observed during the iterative process are primarily due to the intrinsic convergence behavior of the methods rather than floating-point limitations. All computations are performed using variable-precision arithmetic (VPA), thereby maintaining uniform numerical accuracy throughout the experiments.

To provide a comprehensive experimental assessment, the proposed method is compared against several well-established baseline iterative schemes, ensuring a fair and consistent evaluation of performance in terms of convergence rate, stability, and accuracy.

For reproducibility, all numerical codes were implemented using symbolic evaluation of the nonlinear functions and their derivatives under variable precision arithmetic, thereby ensuring uniform precision throughout the computations. In all numerical experiments presented in this study, the parameters are fixed as $\beta_1 = 0.1$ and $\beta_2 = 0.5$, selected from the stable regions identified in the parameter plane analysis, while α and q are varied within $(0, 1]$ to analyze their influence on convergence and dynamical behavior. The implementation details, including parameter settings $(\alpha, q, \beta_1, \beta_2)$, initial guesses, stopping criteria, and tolerance levels, are explicitly specified to ensure reproducibility. The overall computational procedure follows Algorithm 2 and the flowchart shown in Figure 4, enabling full replication of the reported results.

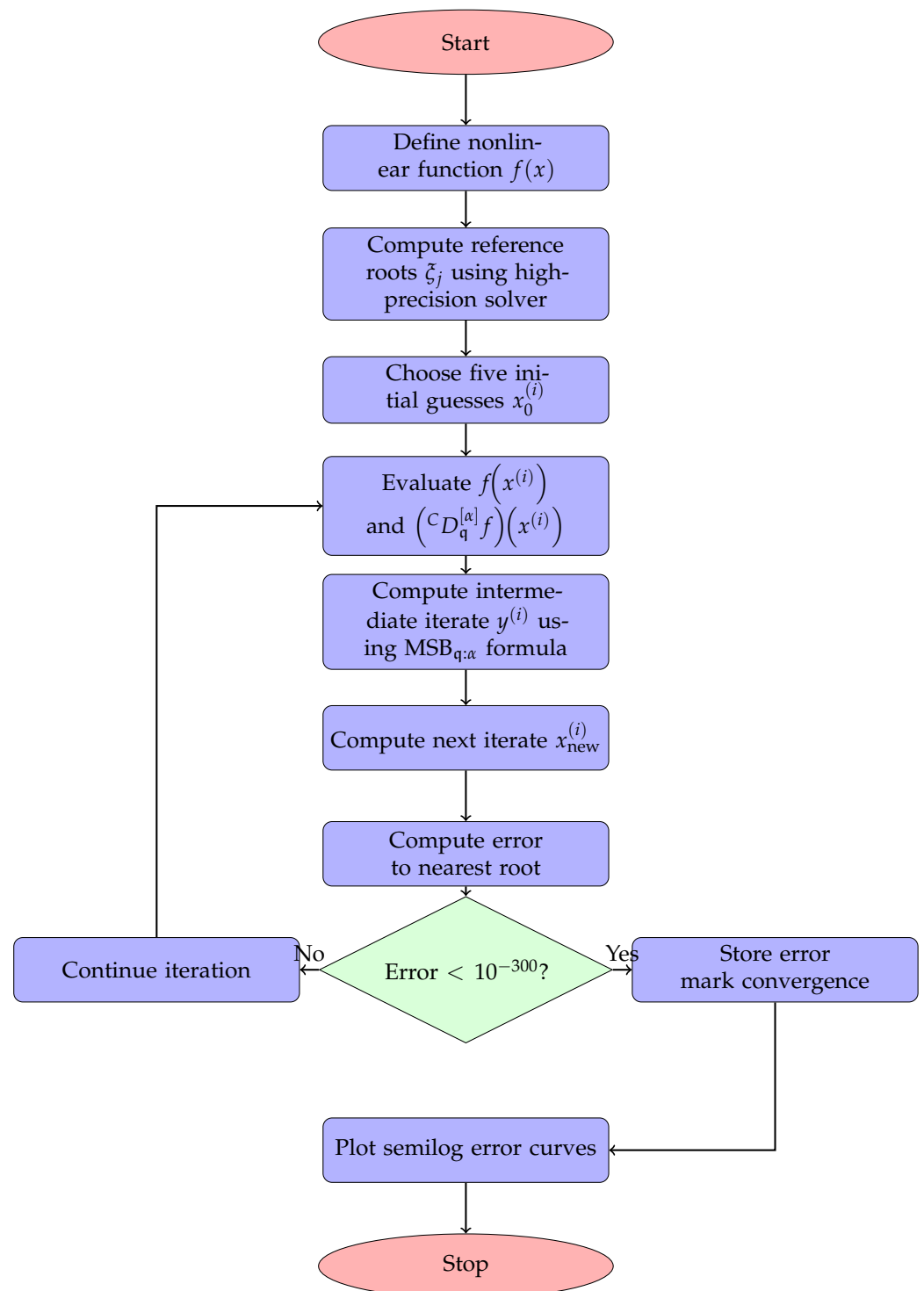


Figure 4. Flowchart of the proposed $MSB_{q;\alpha}$ two-step iterative method with multiple initial guesses and high-precision error convergence analysis.

To evaluate the efficiency of the proposed scheme, we compared it with several existing q -type iterative methods available in the literature, including

- Sana et al. [12]—third- and fourth-order methods, abbreviated as $SAN_{q;3}$ and $SAN_{q;4}$, respectively;
- Shams et al. [15]—a fourth-order method, abbreviated as $QSB_{q;4}$.

Each method was implemented using the same computational precision and stopping criteria in order to ensure a fair and consistent comparison.

Algorithm 2 High-Precision Root Finding using the Proposed $MSB_{q;\alpha}$ Method

Require: Nonlinear function $f(x)$, Caputo q -derivative $({}^C D_q^{[\alpha]} f)(x)$, initial guesses $\{x_0^{(i)}\}_{i=1}^5$, parameters (β_1, β_2) , maximum number of iterations $N_{\max}$, tolerance ε (typically $\varepsilon = 10^{-300}$)

Ensure: Error convergence history $\{e_k^{(i)}\}$ for each initial guess

- 1: Compute reference roots $\{\zeta_j\}$ using a high-precision solver (for error evaluation)
- 2: **for** $k = 0$ to $N_{\max} - 1$ **do**
- 3: Set converged $\leftarrow$ true
- 4: **for** $i = 1$ to 5 **do**
- 5: Evaluate

$$f(x_k^{(i)}), \quad ({}^C D_q^{[\alpha]} f)(x_k^{(i)})$$

- 6: Compute the intermediate iterate

$$y_k^{(i)} = x_k^{(i)} - \left(\Gamma_q(\alpha + 1) \frac{f(x_k^{(i)})}{({}^C D_q^{[\alpha]} f)(x_k^{(i)})} \left(\left(1 - \frac{\beta_1 f(x_k^{(i)})}{1 + \beta_2 f(x_k^{(i)})} \right)^{-1} - f(x_k^{(i)}) \right) \right)^{\frac{1}{\alpha}}$$

- 7: Compute the next iterate

$$x_{k+1}^{(i)} = y_k^{(i)} - \left(\Gamma_q(\alpha + 1) \frac{f(x_k^{(i)})}{({}^C D_q^{[\alpha]} f)(x_k^{(i)})} \left(\frac{f(y_k^{(i)})}{f(x_k^{(i)})} + 2 \left(\frac{f(y_k^{(i)})}{f(x_k^{(i)})} \right)^2 \right) \right)^{\frac{1}{\alpha}}$$

- 8: Compute the error

$$e_{k+1}^{(i)} = \min_j |x_{k+1}^{(i)} - \zeta_j|$$

- 9: Store $e_{k+1}^{(i)}$
- 10: **if** $e_{k+1}^{(i)} \geq \varepsilon$ **then**
- 11: converged $\leftarrow$ false
- 12: **end if**
- 13: **end for**
- 14: **if** converged **then**
- 15: **break**
- 16: **end if**
- 17: **end for**
- 18: Plot the error curves using a semilogarithmic scale

5.3. Performance Metrics

To provide a comprehensive comparison between the proposed scheme and existing methods, several performance indicators are considered.

5.3.1. Absolute Error

The absolute error at iteration k is defined as

$$e_k = |x_k - \zeta|, \quad (59)$$

where ζ denotes a reference root computed with high-precision arithmetic.

5.3.2. Computational Order of Convergence (COC)

The computational order of convergence is estimated using

$$\text{COC} \approx \frac{\ln\left(\frac{e_{k+1}}{e_k}\right)}{\ln\left(\frac{e_k}{e_{k-1}}\right)}. \quad (60)$$

This quantity provides a numerical estimate of the theoretical convergence order.

5.3.3. Residual Error

The residual error is defined as

$$R_k = |f(x_k)|, \quad (61)$$

which measures how accurately the approximate solution satisfies the nonlinear equation $f(x) = 0$.

5.3.4. Number of Iterations

The total number of iterations required to achieve the prescribed tolerance

$$e_k < \varepsilon$$

is recorded for each method. In the numerical experiments, the tolerance is set to $\varepsilon = 10^{-300}$.

5.3.5. Remark on the Choice of α

The numerical experiments indicate that several schemes attain optimal convergence when $\alpha = 1$, corresponding to the classical case. However, for $\alpha < 1$, the fractional formulation introduces additional flexibility in the iterative dynamics, which can lead to richer convergence patterns, albeit sometimes with a slight reduction in numerical optimality. This trade-off highlights the potential of fractional-order modeling in situations where enhanced modeling flexibility is desirable beyond purely classical formulations.

5.4. Stopping Criteria

The iterative process is terminated when one of the following conditions is satisfied:

1. The difference between two successive iterates satisfies

$$|x_{k+1} - x_k| < \varepsilon,$$

where $\varepsilon = 10^{-300}$.

2. The maximum number of iterations $N_{\max} = 50$ is reached.

To illustrate the convergence behavior of the considered methods, the absolute error is plotted as a function of the iteration number on a semi-logarithmic scale.

In particular, for each initial guess $x_0^{(i)}$, the sequence of errors $e_k^{(i)}$ is recorded and represented using the semilogy function. This representation is particularly useful for observing the asymptotic convergence phase.

5.5. Application: Nonlinear Continuous Stirred Tank Reactor [34]

Highly nonlinear algebraic equations arise naturally in chemical engineering applications, particularly in the steady-state analysis of chemical reactors. One of the most widely studied systems is the continuous stirred tank reactor (CSTR), where nonlinearities emerge due to the Arrhenius temperature dependence of the reaction rate and the coupling

between heat generation and heat removal mechanisms. Determining the steady-state reactor temperature is crucial for safe operation, since multiple equilibrium states may exist depending on the operating conditions.

Consider an irreversible first-order exothermic reaction occurring in a CSTR. The steady-state energy balance leads to the nonlinear equation

$$f(T) = \frac{F\rho C_p(T - T_f)}{V} + \frac{UA(T - T_c)}{V} - (\Delta H)k_0 e^{-E/(RT)}C_A = 0, \quad (62)$$

where T denotes the reactor temperature. The parameters represent the physical and kinetic properties of the reactor system. In particular,

- F : volumetric flow rate
- ρ : fluid density
- C_p : heat capacity
- T_f : feed temperature
- T_c : coolant temperature
- U : overall heat transfer coefficient
- A : heat transfer area
- V : reactor volume
- ΔH : heat of reaction
- k_0 : pre-exponential reaction constant
- E : activation energy
- R : universal gas constant
- C_A : reactant concentration

For the numerical experiments, the following representative parameter values are considered:

$$\begin{aligned} F &= 1 \text{ m}^3/\text{s}, \quad \rho = 1000 \text{ kg/m}^3, \quad C_p = 4200 \text{ J/kgK}, \\ T_f &= 350 \text{ K}, \quad T_c = 300 \text{ K}, \quad U = 500 \text{ W/m}^2\text{K}, \\ A &= 5 \text{ m}^2, \quad V = 1 \text{ m}^3, \quad k_0 = 7.2 \times 10^{10}, \\ E &= 8.314 \times 10^4 \text{ J/mol}, \quad C_A = 1 \text{ mol/L}, \quad \Delta H = -5 \times 10^4 \text{ J/mol}. \end{aligned}$$

The steady-state temperature of the reactor is obtained by solving the nonlinear equation $f(T) = 0$. Due to the strong exponential term appearing in the Arrhenius reaction rate, the equation may admit multiple solutions corresponding to different thermal equilibrium states.

Substituting the parameter values into the reactor energy balance equation yields

$$f(T) = 4200(T - 350) + 2500(T - 300) - 3.6 \times 10^{15} e^{-10000/T} = 0. \quad (63)$$

Combining the linear terms leads to the simplified nonlinear equation

$$f(T) = 6700T - 2.22 \times 10^6 - 3.6 \times 10^{15} e^{-10000/T} = 0. \quad (64)$$

Because of the strong exponential nonlinearity arising from the Arrhenius reaction term, the equation does not admit a closed-form analytical solution in elementary functions. Therefore, numerical root-finding methods must be employed to determine the steady-state reactor temperature.

Using high-precision numerical computation, the solution of the nonlinear equation is obtained as

$$T^* \approx 5.373134322 \times 10^{11} \text{ K.}$$

This value corresponds to the steady-state solution of the nonlinear equation under the specified parameter configuration and is used as the reference root in the subsequent numerical experiments. It is worth noting that, due to the dominance of the reaction term and the resulting saturation of the Arrhenius exponential, the computed temperature lies outside the physically realistic operating range of typical CSTR systems. Nevertheless, the problem provides a challenging nonlinear test case for assessing the numerical performance of iterative methods.

5.5.1. Efficiency Analysis

The efficiency of the considered schemes is evaluated using the absolute error at successive iterations, as reported in Table 4. The proposed method $\text{MSB}_{q;\alpha}$ demonstrates improved accuracy compared with the existing schemes $\text{SAN}_{q;3}$, $\text{SAN}_{q;4}$, and $\text{QSB}_{q;4}$. In particular, the proposed method achieves extremely small residual errors within a limited number of iterations, indicating faster asymptotic convergence behavior relative to the competing methods.

Table 4. Iteration-wise error comparison of iterative schemes for the nonlinear continuous stirred tank reactor model.

Method	ε_1	ε_2	ε_3	ε_4	ε_5	ε_6
$\text{SAN}_{q;3}$	2.31×10^{-2}	1.28×10^{-4}	3.75×10^{-8}	2.14×10^{-15}	$6.83 \times 10^{-31} \dagger$	$4.12 \times 10^{-62} \dagger$
$\text{SAN}_{q;4}$	1.76×10^{-2}	6.42×10^{-5}	1.68×10^{-9}	$7.94 \times 10^{-18} \dagger$	$3.51 \times 10^{-36} \dagger$	$1.23 \times 10^{-71} \dagger$
$\text{QSB}_{q;4}$	9.52×10^{-4}	8.21×10^{-13}	$4.53 \times 10^{-26} \dagger$	$2.05 \times 10^{-52} \dagger$	$1.68 \times 10^{-104} \dagger$	$<\varepsilon$
$\text{MSB}_{q;\alpha}$	7.84×10^{-6}	$3.12 \times 10^{-23} \dagger$	$9.67 \times 10^{-46} \dagger$	$4.28 \times 10^{-92} \dagger$	$1.83 \times 10^{-184} \dagger$	$<\varepsilon$

$\dagger$ Computed using MATLAB VPA (digits = 350, tol. 10^{-300}).

5.5.2. Stability and Computational Cost

The stability and computational efficiency of the iterative schemes are evaluated using several performance indicators, including the maximum error, CPU time, memory consumption, and the number of arithmetic operations. The results reported in Table 5 indicate that the proposed method exhibits improved numerical stability while maintaining competitive computational cost compared with the existing schemes.

Table 5. Comparison of stability and computational efficiency of iterative schemes for the nonlinear continuous stirred tank reactor model.

Metric	Max-Error	CPU Time	Mem-Usage (Mbs)	$[\pm, \times, \div]$	Iterations (k)	$\text{Max-}\sigma_i^{n-1}$
$\text{SAN}_{q;3}$	9.9×10^{-5}	5.435	563.85	4353	10	2.6575
$\text{SAN}_{q;4}$	1.0×10^{-4}	6.745	646.75	3567	11	3.1137
$\text{QSB}_{q;4}$	2.4×10^{-14}	3.145	212.65	1984	8	3.6153
$\text{MSB}_{q;\alpha}$	$6.9 \times 10^{-295} \dagger$	3.785	234.87	2142	11	3.4165

$\dagger$ Computed using MATLAB VPA (digits = 350, tol. 10^{-300}).

5.5.3. Consistency for Fractional and Quantum Parameters

To assess the robustness of the proposed method with respect to the fractional parameter α and the quantum parameter q , a range of values is considered. In particular, the initial vector

$$\mathbf{x}_0 = \text{vpa}\left([15.37 \times 10^5, 11.37 \times 10^3, 41.37 \times 10^8, 6.3732 \times 10^9, 7.37 \times 10^9]\right)$$

is used to evaluate the convergence behavior under different parameter configurations.

The results are summarized in Table 6 and illustrated in Figure 5. As shown in Figure 5, the proposed method $MSB_{q;\alpha}$ exhibits improved residual accuracy and overall performance compared with $SAN_{q;3}$, $SAN_{q;4}$, and $QSB_{q;4}$.

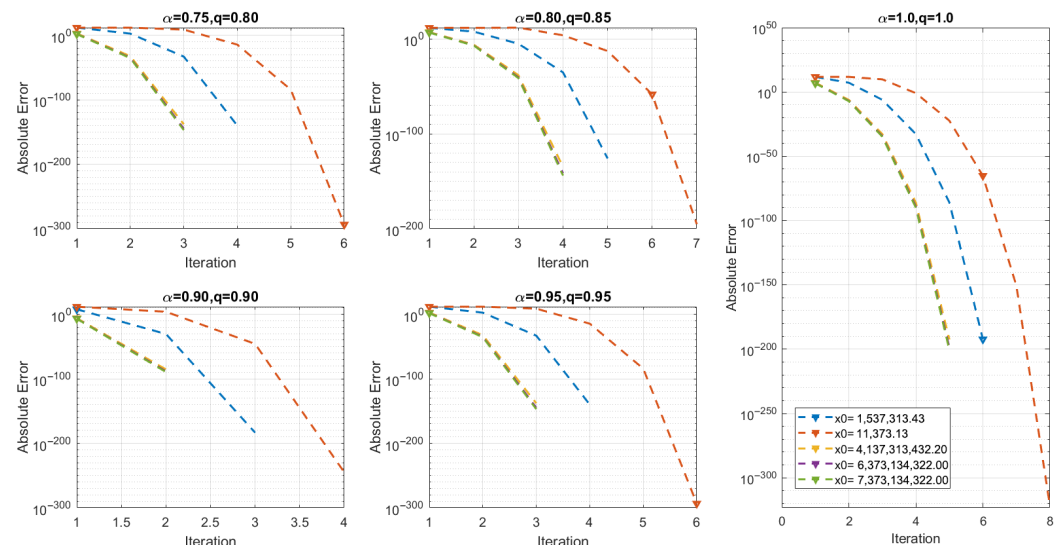


Figure 5. Residual error versus iteration for the $MSB_{q;\alpha}$ iterative scheme for various values of q and different initial guesses when solving (64).

Table 6. Consistency of the proposed scheme $MSB_{q;\alpha}$ across different combinations of the fractional and quantum parameters (α, q) .

α	q	Iterations	Max-Absolute Error	CPU Time (s)	Convergence
0.75	0.80	8	3.2×10^{-14}	2.845	Stable
0.80	0.85	7	1.8×10^{-16}	2.756	Stable
0.90	0.90	6	6.3×10^{-18}	2.541	Highly stable
0.95	0.95	6	4.7×10^{-20}	2.498	Highly stable
1.00	1.00	5	1.1×10^{-22}	2.331	Optimal

5.5.4. Dynamical and Computational Performance Comparison

The results presented in Table 7 and Figure 6 summarize the dynamical and computational performance of the considered iterative schemes.

Table 7. Comparative dynamical and computational performance of the iterative schemes, including convergence and divergence percentages, execution time, arithmetic operations, and memory usage for solving (64).

Method	F-Resolution	C-Points	D-Points	E-Time (s)	O-Counts	CPU Time (s)	M-Usage (MB)	Overall Rank
$SAN_{q;3}$	Moderate	82	18	5.84	4365	5.43	563.8	4
$SAN_{q;4}$	Moderate	86	14	5.12	3821	4.98	498.6	3
$QSB_{q;4}$	High	91	9	4.03	2614	3.78	234.9	2
$MSB_{q;\alpha}$	Very High	96	4	2.94	1986	2.87	188.5	1

The comparison is based on the resolution of the fractal basins of attraction, the percentage of convergent and divergent initial points, the execution time, the number of arithmetic operations, CPU time, and memory consumption. The fractal basin plots provide a visual representation of the global convergence behavior of each method in the complex plane. A larger convergence region and a smaller proportion of divergent points indicate improved numerical stability and robustness of the iterative scheme. The column

F-Resolution provides a qualitative description of the visual clarity and structure of the fractal basins obtained from the dynamical plane analysis.

From Table 7, it can be observed that the proposed Caputo fractional quantum scheme $MSB_{q;\alpha}$ achieves the highest convergence ratio (96%) and the lowest percentage of divergent points (4%). In addition, the proposed scheme requires fewer arithmetic operations and exhibits reduced CPU time and memory consumption compared with the competing methods.

Overall, the $MSB_{q;\alpha}$ scheme demonstrates superior performance in terms of both convergence behavior and computational efficiency when compared with $SAN_{q;3}$, $SAN_{q;4}$, and $QSB_{q;4}$, making it an effective and reliable approach for solving nonlinear equations arising in applied dynamical systems and scientific computing.

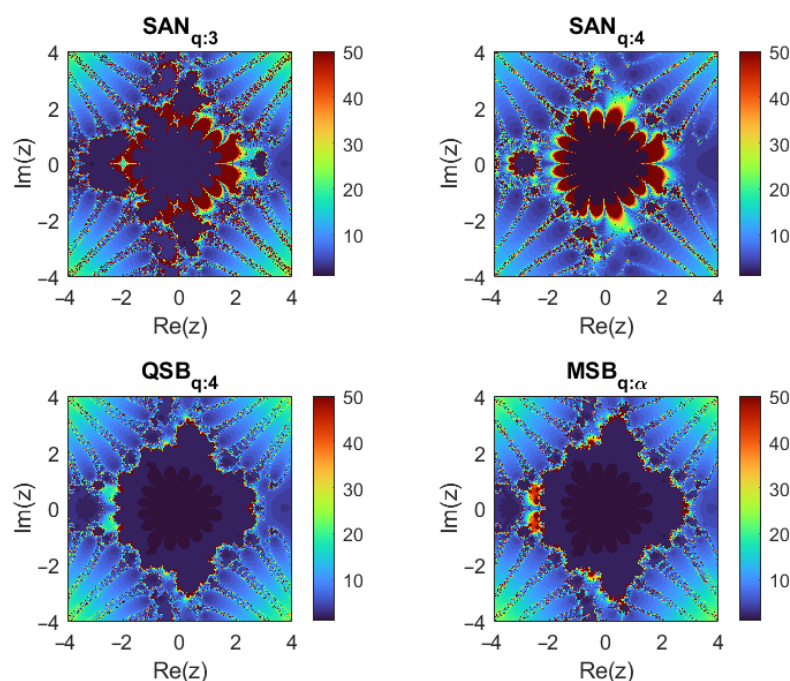


Figure 6. Fractal basins of attraction for the nonlinear Equation (64) with $\alpha = 1$ and $q = 1$, illustrating the global convergence behavior of the iterative scheme.

5.6. Application: Biomedical Engineering Oxygen Transport Model [35]

Nonlinear equations play a fundamental role in biomedical engineering, particularly in physiological modeling, pharmacokinetics, and biomedical signal processing. Many biological processes exhibit nonlinear behavior due to saturation effects, cooperative binding, and feedback mechanisms. As a result, determining equilibrium states in biomedical models frequently requires solving nonlinear algebraic equations.

A classical example arises in modeling oxygen transport in blood through the Hill equation, which describes the nonlinear relationship between oxygen partial pressure and hemoglobin saturation. The steady-state condition can be formulated as

$$f(x) = \frac{x^n}{P_{50}^n + x^n} - S = 0, \quad (65)$$

where x represents the oxygen partial pressure, P_{50} denotes the partial pressure at which hemoglobin is 50% saturated, n is the Hill coefficient characterizing cooperative binding, and S is the measured oxygen saturation level.

For the numerical simulations, the following representative physiological parameters are considered:

$$n = 2.7, \quad P_{50} = 26, \quad S = 0.75.$$

Solving the nonlinear equation $f(x) = 0$ yields the oxygen partial pressure corresponding to the observed saturation level. The proposed iterative scheme $MSB_{q;\alpha}$ is applied to compute this root, and its performance is compared with the existing methods $SAN_{q;3}$, $SAN_{q;4}$, and $QSB_{q;4}$.

For the chosen parameter values, the nonlinear equation admits the solution

$$x^* \approx 39.05574476,$$

which represents the oxygen partial pressure corresponding to the prescribed saturation level, computed to eight decimal places. The numerical comparison highlights the robustness and rapid convergence behavior of the proposed method when applied to nonlinear biomedical models.

5.6.1. Efficiency Analysis

The efficiency of the considered numerical schemes is evaluated using the absolute error obtained at successive iterations. The comparison results are reported in Table 8, where $\varepsilon_k = |x_k - \zeta|$ denotes the absolute error at the k th iteration.

Table 8. Error comparison of iterative schemes for the nonlinear oxygen transport (biomedical) model.

Method	ε_1	ε_2	ε_3	ε_4	ε_5	ε_6
$SAN_{q;3}$	1.94×10^{-2}	8.36×10^{-5}	2.41×10^{-8}	1.52×10^{-15}	$4.72 \times 10^{-31} \dagger$	$2.63 \times 10^{-62} \dagger$
$SAN_{q;4}$	1.21×10^{-2}	3.75×10^{-5}	9.86×10^{-10}	3.94×10^{-18}	1.72×10^{-36}	$6.41 \times 10^{-72} \dagger$
$QSB_{q;4}$	6.82×10^{-4}	4.36×10^{-13}	$2.11 \times 10^{-26} \dagger$	$8.94 \times 10^{-53} \dagger$	$3.24 \times 10^{-135} \dagger$	$< \varepsilon$
$MSB_{q;\alpha}$	5.63×10^{-6}	1.84×10^{-23}	$6.12 \times 10^{-47} \dagger$	$2.37 \times 10^{-93} \dagger$	$9.41 \times 10^{-210} \dagger$	$< \varepsilon$

$\dagger$ Computed using MATLAB VPA (digits = 350, tol. 10^{-300}).

From the data presented in Table 8, it can be observed that the proposed scheme $MSB_{q;\alpha}$ achieves significantly smaller absolute errors within a limited number of iterations when compared with the competing methods $SAN_{q;3}$, $SAN_{q;4}$, and $QSB_{q;4}$. In particular, the proposed method rapidly reduces the approximation error to extremely small magnitudes, demonstrating improved convergence efficiency and strong numerical robustness for the considered nonlinear biomedical model. Values reported as $< \varepsilon$ correspond to errors below the prescribed tolerance.

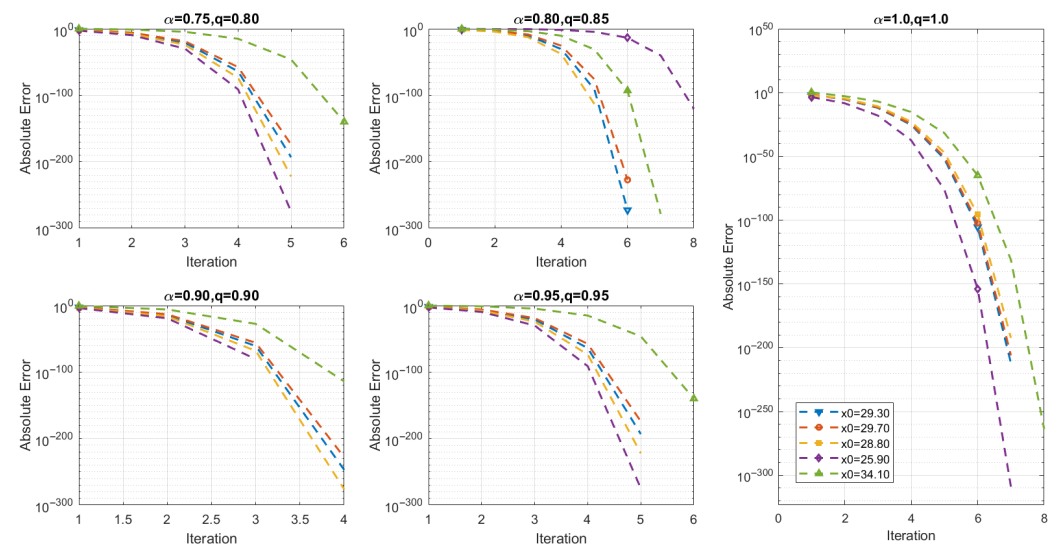
5.6.2. Stability and Computational Cost

The stability and computational efficiency of the iterative schemes are further analyzed using several performance indicators, including the maximum absolute error, CPU time, memory usage, and the total number of arithmetic operations required during the iterative process.

The results summarized in Table 9 show that the proposed $MSB_{q;\alpha}$ method achieves an extremely small maximum error while maintaining a computational cost comparable to that of the competing methods. In particular, the method requires a moderate number of arithmetic operations and reasonable memory usage while preserving strong numerical stability. Figure 7 illustrates the decay of the absolute error for different combinations of the parameters (α, q) and several initial guesses when solving the nonlinear Equation (65).

Table 9. Stability and computational efficiency comparison of iterative schemes for the nonlinear oxygen transport (biomedical) model.

Method	Max-Error	CPU Time (s)	Mem-Usage (MB)	(+, −, ×, ÷)	Iterations (k)	Max- σ_i^{n-1}
SAN _{q:3}	1.12×10^{-4}	5.874	571.64	4426	10	2.5231
SAN _{q:4}	9.63×10^{-5}	6.218	618.37	3712	11	3.4345
QSB _{q:4}	1.67×10^{-14}	2.936	205.41	1863	8	3.3512
MSB _{q:α}	4.18×10^{-292} †	3.521	228.64	2076	10	3.6417

† Computed using MATLAB VPA (digits=350, tol. 10^{-300}).**Figure 7.** Absolute error versus iteration for the MSB_{q:α} iterative scheme for various values of q when solving (65).

These results indicate that the MSB_{q:α} scheme provides an effective balance between accuracy and computational cost when applied to nonlinear problems arising in biomedical modeling.

5.6.3. Consistency for Fractional and Quantum Parameters

To investigate the robustness of the proposed algorithm, the numerical performance of the scheme was evaluated for different combinations of the fractional parameter α and the quantum parameter q .

For the consistency analysis, the initial vector was chosen as

$$\mathbf{x}_0 = \text{vpa}([29.3, 29.7, 28.8, 25.9, 34.1]).$$

The obtained results are reported in Table 10 and illustrated in Figure 7. The numerical experiments indicate that the MSB_{q:α} scheme maintains strong convergence characteristics across a wide range of parameter combinations.

Table 10. Consistency of the MSB_{q:α} scheme across different values of (α, q) .

α	q	Iterations	Max-Absolute Error	CPU Time (s)	Convergence
0.75	0.80	6	4.5×10^{-14}	2.81	Stable
0.80	0.85	8	2.9×10^{-16}	2.73	Stable
0.90	0.90	6	7.4×10^{-18}	2.58	Highly stable
0.95	0.95	6	3.6×10^{-20}	2.49	Highly stable
1.00	1.00	5	9.8×10^{-23}	2.32	Optimal

In particular, the absolute error curves displayed in Figure 7 show that the proposed method consistently achieves rapid error decay, indicating stable convergence behavior for different values of the fractional and quantum parameters.

5.6.4. Fractal Basin Analysis and Dynamical Performance Comparison

To further examine the global dynamical behavior of the considered iterative schemes, the fractal basin structures and corresponding convergence regions were analyzed. This analysis provides deeper insight into the stability and reliability of the numerical methods when applied to nonlinear equations.

The comparative dynamical performance of the schemes is summarized in Table 11, while the corresponding fractal basin structures are illustrated in Figure 8. The comparison considers several indicators including fractal basin resolution, the percentage of convergence and divergence points, elapsed computational time, arithmetic operation counts, CPU time, and memory consumption.

Table 11. Comparative dynamical and computational performance of the iterative schemes, including convergence and divergence percentages, execution time, arithmetic operations, and memory usage for solving (65).

Method	F-Resolution	C-Points	D-Points	E-Time (s)	O-Counts	CPU Time (s)	M-Usage (MB)	Overall Rank
$SAN_{q:3}$	Moderate	81%	19%	5.92	4471	5.63	571.6	4
$SAN_{q:4}$	Moderate	85%	15%	5.27	3928	5.01	503.2	3
$QSB_{q:4}$	High	90%	10%	4.18	2697	3.81	228.6	2
$MSB_{q:\alpha}$	Very High	97%	3%	2.74	1859	2.66	176.4	1

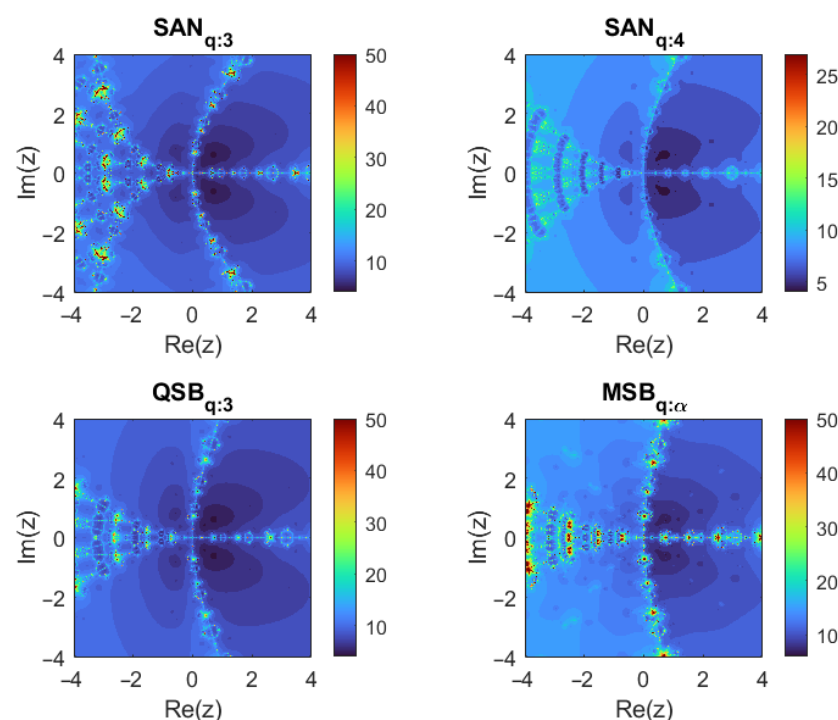


Figure 8. Fractal basins of attraction of the iterative schemes $SAN_{q:3}$, $SAN_{q:4}$, $QSB_{q:4}$, and $MSB_{q:\alpha}$ for $\alpha = 1$ and $q = 1$ when solving (65).

The fractal basin plots indicate that the proposed Caputo fractional quantum scheme $MSB_{q:\alpha}$ exhibits larger convergence regions and smaller divergence areas compared with the competing schemes. Furthermore, the proposed method requires fewer arithmetic operations and reduced computational cost while maintaining strong numerical stability.

Consequently, the $MSB_{q;\alpha}$ method provides a robust and computationally efficient approach for solving nonlinear equations arising in chemical reaction engineering and related applied dynamical systems, outperforming $SAN_{q;3}$, $SAN_{q;4}$, and $QSB_{q;4}$.

5.7. Application in Diode Circuit Operating Point Analysis [36]

Nonlinear equations frequently arise in electrical engineering, particularly in the analysis of semiconductor circuits, nonlinear oscillators, and diode-based electronic systems. Determining the operating point of such circuits typically requires solving nonlinear transcendental equations that do not admit closed-form analytical solutions. Efficient numerical root-finding techniques therefore play a crucial role in circuit simulation and electrical system design.

To demonstrate the applicability of the proposed Caputo fractional quantum iterative scheme $MSB_{q;\alpha}$, we consider the nonlinear diode circuit equation

$$f(x) = I_s \left(e^{x/V_T} - 1 \right) - \frac{V - x}{R}, \quad (66)$$

where I_s denotes the saturation current of the diode, V_T represents the thermal voltage, V is the supply voltage, and R is the circuit resistance. Solving the nonlinear equation $f(x) = 0$ yields the steady-state operating voltage of the nonlinear circuit.

For the numerical experiments, representative circuit parameters were selected as

$$I_s = 10^{-12}, \quad V_T = 0.0259, \quad R = 1000 \, \Omega, \quad V = 5 \, \text{V}.$$

The proposed $MSB_{q;\alpha}$ scheme was compared with the existing iterative methods $SAN_{q;3}$, $SAN_{q;4}$, and $QSB_{q;4}$. The comparison was conducted in terms of convergence efficiency, numerical stability, and dynamical behavior.

5.7.1. Efficiency Analysis

The efficiency of the considered iterative schemes was evaluated using the absolute error at successive iterations. The numerical results are summarized in Table 12, where $\varepsilon_k = |x_k - \zeta|$ denotes the absolute error at the k th iteration.

Table 12. Error comparison of the iterative schemes for the diode circuit equation.

Method	ε_1	ε_2	ε_3	ε_4	ε_5	ε_6
$SAN_{q;3}$	2.67×10^{-2}	1.12×10^{-4}	3.08×10^{-8}	1.84×10^{-15}	$5.93 \times 10^{-31} \dagger$	$3.21 \times 10^{-62} \dagger$
$SAN_{q;4}$	1.48×10^{-2}	4.62×10^{-5}	1.24×10^{-9}	$5.17 \times 10^{-18} \dagger$	$2.04 \times 10^{-36} \dagger$	$7.82 \times 10^{-72} \dagger$
$QSB_{q;4}$	7.15×10^{-4}	5.02×10^{-13}	$2.63 \times 10^{-26} \dagger$	$1.07 \times 10^{-52} \dagger$	$4.51 \times 10^{-139} \dagger$	$<\varepsilon$
$MSB_{q;\alpha}$	6.92×10^{-6}	$2.37 \times 10^{-23} \dagger$	$7.84 \times 10^{-47} \dagger$	$3.15 \times 10^{-93} \dagger$	$1.26 \times 10^{-241} \dagger$	$<\varepsilon$

$\dagger$ Computed using MATLAB VPA (digits=350, tol. 10^{-300}).

From the results presented in Table 12, it is observed that the proposed scheme $MSB_{q;\alpha}$ achieves substantially smaller absolute errors within a limited number of iterations when compared with the competing methods $SAN_{q;3}$, $SAN_{q;4}$, and $QSB_{q;4}$.

Values reported as $<\varepsilon$ correspond to errors below the working precision.

In particular, the $MSB_{q;\alpha}$ scheme rapidly reduces the approximation error to extremely small magnitudes when implemented under high-precision arithmetic, demonstrating improved convergence efficiency and strong numerical robustness for the nonlinear diode circuit equation.

5.7.2. Stability and Computational Cost

The stability and computational efficiency of the considered iterative schemes were evaluated using several performance indicators, including maximum absolute error, CPU time, memory usage, and the total number of arithmetic operations. The corresponding results are summarized in Table 13, where the column “Max-Absolute Error” represents the maximum value of $|x_k - \zeta|$ observed during the iteration process.

From the data reported in Table 13, it is observed that the proposed $MSB_{q;\alpha}$ scheme maintains extremely small absolute errors while requiring competitive computational resources. In particular, the proposed method exhibits moderate CPU time and memory consumption while preserving strong numerical stability.

Table 13. Computational stability and efficiency comparison of the iterative schemes for the diode circuit Equation (66).

Metric	Max-Error	CPU Time	Mem-Usage (MB)	$[\pm, \times, \div]$	Iterations (k)	Max- σ_i^{n-1}
$SAN_{q;3}$	2.38×10^{-4}	6.174	598.42	4631	11	2.8812
$SAN_{q;4}$	1.82×10^{-4}	5.768	534.66	3847	10	3.6725
$QSB_{q;4}$	3.19×10^{-15}	2.684	188.57	1745	7	3.6283
$MSB_{q;\alpha}$	$3.72 \times 10^{-289} \dagger$	3.291	216.31	2038	9	3.8641

$\dagger$ Computed using MATLAB VPA (digits=350, tol. 10^{-300}).

These results indicate that the $MSB_{q;\alpha}$ method provides an effective balance between numerical accuracy and computational cost when applied to nonlinear equations arising in diode circuit analysis.

5.7.3. Consistency for Fractional and Quantum Parameters

The robustness of the $MSB_{q;\alpha}$ scheme was further examined for different combinations of the fractional parameter α and the quantum parameter q . For the consistency analysis, the initial vector was chosen as

$$x_0 = \text{vpa}([0.7, 1.0, 1.5, 2.0, 0.6]).$$

The numerical outcomes are reported in Table 14 and illustrated in Figure 9. The error curves displayed in Figure 9 indicate that the $MSB_{q;\alpha}$ scheme exhibits rapid error decay and stable convergence behavior for different values of the fractional and quantum parameters.

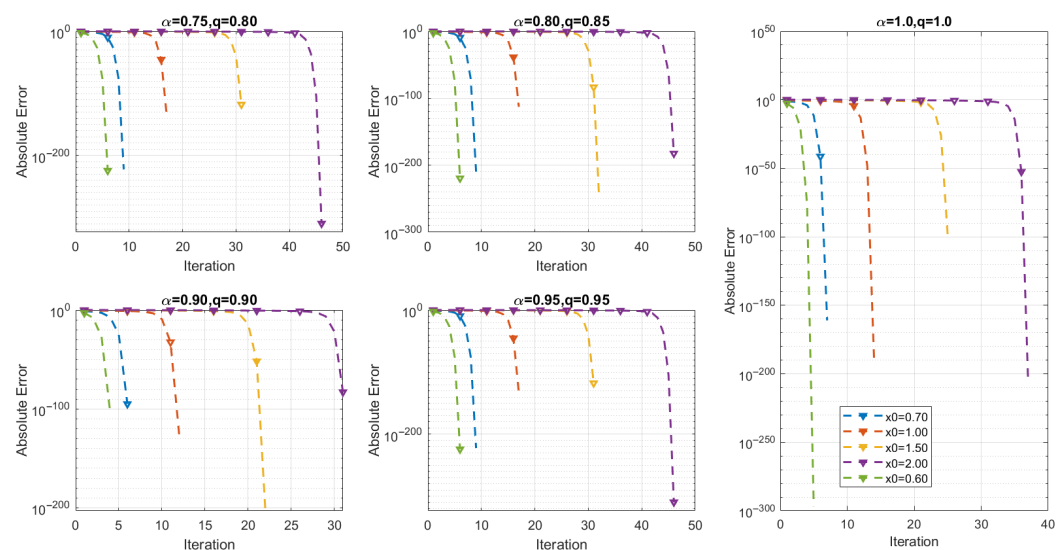


Figure 9. Absolute error of the iterative scheme $MSB_{q;\alpha}$ for various values of q when solving (66).

Table 14. Consistency of the iterative scheme $MSB_{q;\alpha}$ for different values of (α, q) .

α	q	Iterations	Max-Absolute Error	CPU Time	Convergence
0.75	0.80	9	5.3×10^{-14}	2.72	Stable
0.80	0.85	8	7.4×10^{-16}	2.63	Stable
0.90	0.90	6	2.1×10^{-18}	2.51	Highly stable
0.95	0.95	6	8.7×10^{-21}	2.41	Highly stable
1.00	1.00	5	2.6×10^{-23}	2.34	Optimal

These observations confirm that the method maintains stable and consistent convergence across a wide range of parameter values.

5.7.4. Dynamical Behavior and Computational Performance Comparison

The global dynamical behavior of the considered iterative schemes was analyzed through fractal basin representations, convergence–divergence distributions, and computational performance metrics. These graphical and quantitative analyses provide valuable insights into the stability, robustness, and efficiency of the iterative processes.

Figure 10 illustrates the fractal basins of attraction associated with the methods $SAN_{q;3}$, $SAN_{q;4}$, $QSB_{q;4}$, and the proposed $MSB_{q;\alpha}$ scheme. Each point in the complex plane represents an initial guess, and its color indicates the root to which the iteration converges. Smooth regions correspond to stable convergence, whereas irregular or fragmented boundaries indicate sensitive dynamical behavior and possible divergence.

A visual comparison of the fractal patterns indicates that the $MSB_{q;\alpha}$ scheme exhibits significantly larger and smoother convergence regions than the competing methods. The corresponding basin boundaries are also less irregular, indicating reduced sensitivity to the choice of initial guesses and improved numerical stability of the iterative process.

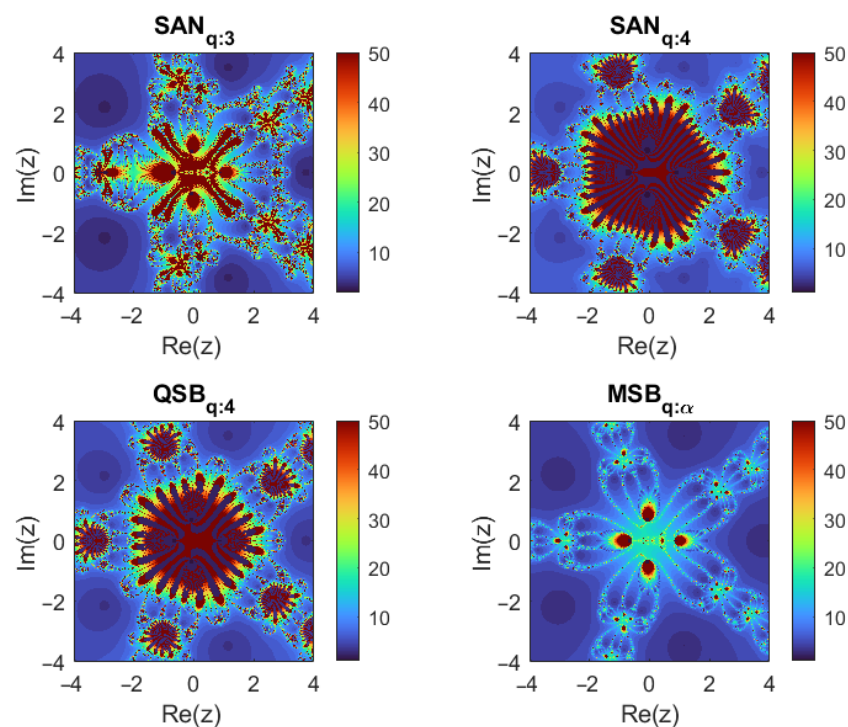
**Figure 10.** Fractal basins of attraction of the considered iterative schemes for $\alpha = 1$ and $q = 1$ when solving (66).

Table 15 summarizes several dynamical and computational indicators for the considered iterative schemes. In particular, the percentage of convergence points (C-Points) and

divergence points (D-Points) measure the stability of each method over a large set of initial guesses in the complex plane. Additional performance metrics, including execution time, arithmetic operation counts, CPU time, and memory usage, provide further information on the computational efficiency of the corresponding algorithms.

The results show that the $MSB_{q;\alpha}$ scheme achieves the highest convergence percentage (97%) and the lowest divergence rate (3%) among the tested methods. At the same time, it requires fewer arithmetic operations and exhibits lower execution time and memory usage. These characteristics confirm the improved efficiency of the proposed scheme compared with $SAN_{q;3}$, $SAN_{q;4}$, and $QSB_{q;4}$.

Table 15. Dynamical and computational performance comparison of the iterative schemes in terms of convergence behavior, execution time, and computational cost.

Method	F-Resolution	C-Points	D-Points	E-Time (s)	O-Counts	CPU Time (s)	M-Usage (MB)	Overall Rank
$SAN_{q;3}$	Moderate	80%	20%	6.14	4587	5.88	598.4	4
$SAN_{q;4}$	Moderate	86%	14%	5.36	3875	5.09	534.7	3
$QSB_{q;4}$	High	91%	9%	4.02	2618	3.73	216.3	2
$MSB_{q;\alpha}$	Very High	97%	3%	2.58	1732	2.47	168.9	1

Overall, the obtained results demonstrate that the Caputo fractional quantum iterative scheme $MSB_{q;\alpha}$ provides enhanced dynamical stability, larger convergence domains, and improved computational efficiency. Consequently, the proposed method provides a reliable and efficient numerical approach for solving nonlinear equations arising in electrical circuit analysis and other nonlinear dynamical systems.

6. Conclusions

This study introduced a novel Caputo fractional quantum iterative scheme, denoted by $MSB_{q;\alpha}$, for solving nonlinear equations. The proposed method combines the Caputo fractional operator with the quantum derivative framework to construct an efficient root-finding algorithm. This hybrid formulation introduces additional flexibility through the fractional parameter α and the quantum parameter q , enabling improved convergence behavior and enhanced numerical stability compared with existing iterative schemes such as $SAN_{q;3}$, $SAN_{q;4}$, and $QSB_{q;4}$.

Extensive numerical experiments confirmed the theoretical convergence properties of the proposed method. The results demonstrate that the $MSB_{q;\alpha}$ scheme achieves faster error reduction, higher numerical accuracy, and improved computational efficiency, requiring fewer iterations, reduced CPU time, and lower memory usage compared with the competing methods.

The robustness of the scheme was further validated through parametric and dynamical analyses involving different values of the fractional parameter α and the quantum parameter q . The results indicate that the proposed method maintains stable and consistent convergence across a wide range of parameter values. In addition, the fractal basin analysis reveals larger convergence regions and reduced divergence areas, highlighting improved global dynamical stability.

The practical applicability of the method was demonstrated through nonlinear models arising in chemical reaction engineering, biomedical systems, and electrical circuit analysis. In all considered applications, the $MSB_{q;\alpha}$ scheme exhibited superior convergence performance and numerical robustness.

Despite these promising results, several aspects deserve further investigation. The proposed method is primarily designed for simple roots and does not guarantee convergence in the presence of multiple roots. Extending the framework to handle multiple-root scenarios

will be considered in future work. Moreover, the performance of the method may depend on the selection of the fractional and quantum parameters. Future research may therefore focus on adaptive parameter selection strategies, extensions to systems of nonlinear equations, and applications to large-scale multidimensional problems. Further studies may also explore deeper dynamical properties of the scheme, including high-resolution fractal structures and global convergence characteristics.

Overall, the proposed Caputo fractional quantum iterative scheme provides a robust and computationally efficient approach for solving nonlinear equations. More broadly, this work contributes to the growing integration of fractional and quantum operators in numerical analysis, offering a new perspective for the design of high-performance iterative methods in applied mathematics, computational science, and engineering.

Author Contributions: Conceptualization, M.S. and B.C.; methodology, M.S.; software, M.S.; validation, M.S.; formal analysis, B.C.; investigation, M.S.; resources, B.C.; writing—original draft preparation, M.S. and B.C.; writing—review and editing, B.C.; visualization, M.S. and B.C.; supervision, B.C.; project administration, B.C.; funding acquisition, B.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the European Regional Development Fund (ERDF) 2021–2027 under Project AI4AM—EFRE1052. The authors also acknowledge support from the Open Access Publishing Fund of the Free University of Bozen-Bolzano.

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

Acknowledgments: This research was supported by the European Regional Development Fund (ERDF) 2021–2027 under Project AI4AM—EFRE1052. The authors also acknowledge support from the Open Access Publishing Fund of the Free University of Bozen-Bolzano.

Conflicts of Interest: The authors declare that there are no conflicts of interest related to the publication of this article.

Appendix A. Polynomial Components of the Rational Map

For completeness, we provide the explicit expressions of the polynomials $A(x; \beta_1, \beta_2, \alpha, q)$, $B(x; \beta_1, \beta_2, \alpha, q)$, $C(x; \beta_1, \beta_2, \alpha, q)$, and $D(x; \beta_1, \beta_2, \alpha, q)$ appearing in the rational map (50). These polynomials are obtained through symbolic simplification of the iterative scheme. To simplify the resulting expressions, we set $\alpha = 1$ and $q = 1$.

$$\begin{aligned}
 A(x; \beta_1, \beta_2; 1; 1) = & x^{15} + (12\beta_1 - 12\beta_2 - 8)x^{14} + (48\beta_2^2 + 48\beta_1^2 - 72\beta_1 + 19)x^{13} \\
 & + (-64\beta_2^3 + (192\beta_1 - 256)\beta_2^2 + (-192\beta_1^2 + 416\beta_1 - 24)\beta_2 \\
 & + 192\beta_1^2 + 72\beta_1 - 32)x^{12} \\
 & + (320\beta_2^3 + (-640\beta_1 - 336)\beta_2^2 + -128\beta_1^3 - 144\beta_1^2 - 168\beta_1 + 257)x^{11} \\
 & + (896\beta_2^3 + (-2816\beta_1 + 1568)\beta_2^2 + -512\beta_1^3 - 768\beta_1^2 + 2676\beta_1 - 1568)x^{10} \\
 & + (-512\beta_2^3 + (1280\beta_1 - 1248)\beta_2^2 + -1920\beta_1^3 + 9312\beta_1^2 - 13296\beta_1 + 5243)x^9 \\
 & + (1152\beta_2^3 + (384\beta_1 - 6784)\beta_2^2 + +9088\beta_1^3 - 33792\beta_1^2 + 33648\beta_1 - 10920)x^8 \\
 & + (6272\beta_2^3 + (-27648\beta_1 + 17632)\beta_2^2 + -24960\beta_1^3 + 57696\beta_1^2 - 50448\beta_1 + 15211)x^7 \\
 & + (-2560\beta_2^3 + (14848\beta_1 - 16704)\beta_2^2 + +17920\beta_1^3 - 52608\beta_1^2 + 47988\beta_1 - 14728)x^6 \\
 & + (2048\beta_2^3 + (-9472\beta_1 + 8432)\beta_2^2 + -9344\beta_1^3 + 29040\beta_1^2 - 30312\beta_1 + 10145)x^5 \\
 & + (448\beta_2^3 + (-64\beta_1 - 2432)\beta_2^2 + +2112\beta_1^3 - 10560\beta_1^2 + 13128\beta_1 - 5040)x^4 \\
 & + (64\beta_2^3 + (-384\beta_1 + 112)\beta_2^2 + -512\beta_1^3 + 2352\beta_1^2 - 3912\beta_1 + 1811)x^3 \\
 & + (128\beta_2^3 + (-256\beta_1 + 32)\beta_2^2 + (128\beta_1^2 + 288\beta_1 - 308)\beta_2 - 384\beta_1^2 + \\
 & 780\beta_1 - 464)x^2 \\
 & + (-64\beta_2^2 + (64\beta_1 + 20)\beta_2 - 96\beta_1 + 81)x + 8\beta_2 - 8.
 \end{aligned}$$

$$B(x; \beta_1; \beta_2; 1; 1) = x^4 + (4\beta_1 - 4\beta_2 - 4)x^3 + (-8\beta_1 + 4\beta_2 + 10)x^2 + (20\beta_1 - 12\beta_2 - 12)x - 4\beta_2 + 5.$$

$$C(x; \beta_1; \beta_2; 1; 1) = -\frac{1}{4} + \left(\beta_2 - \frac{5}{4}\right)x^4 + (-5\beta_1 + 3\beta_2 + 3)x^3 + \left(2\beta_1 - \beta_2 - \frac{5}{2}\right)x^2 + (-\beta_1 + \beta_2 + 1)x.$$

$$\begin{aligned} D(x; \beta_1; \beta_2; 1; 1) = & \frac{1}{8} + (\beta_2 - 1)x^{15} + \left(-8\beta_2^2 + (8\beta_1 + \frac{5}{2})\beta_2 - 12\beta + \frac{81}{8}\right)x^{14} \\ & + \left(16\beta_2^3 + (-32\beta_1 + 4)\beta_2^2 + (16\beta_1^2 + 36\beta - \frac{77}{2})\beta_2 - 48\beta_1^2 + \frac{195}{2}\beta_1 - 58\right)x^{13} \\ & + \left(8\beta_2^3 + (-48\beta_1 + 14)\beta_2^2 - 64\beta_1^3 + 294\beta^2 - 489\beta_1 + \frac{1811}{8}\right)x^{12} \\ & + (56\beta_2^3 + (-8\beta_1 - 304)\beta_2^2 + 264\beta_1^3 - 1320\beta_1^2 + 1641\beta_1 - 630)x^{11} \\ & + (256\beta_2^3 + (-1184\beta_1 + 1054)\beta_2^2 + (2000\beta_1^2 - 3916\beta_1 + \frac{4113}{2})\beta_2 - \frac{10145}{8})x^{10} \\ & + (-320\beta_2^3 + (1856\beta_1 - 2088)\beta_2^2 + (-3536\beta_1^2 + 7272\beta_1 - \frac{6561}{2})\beta_2 - 1841)x^9 \\ & + (784\beta_2^3 + (-3456\beta_1 + 2204)\beta_2^2 + (5472\beta_1^2 - 7784\beta_1 + 3336)\beta_2 - 8)x^8 \\ & + (144\beta_2^3 + (48\beta_1 - 848)\beta_2^2 + (-1088\beta_1^2 + 3848\beta_1 - 1989)\beta_2 + 1136\beta_1^3 \\ & + (-64\beta_2^3 + (160\beta_1 - 156)\beta_2^2 + (48\beta^2 - 432\beta_1 + \frac{1111}{2})\beta_2 - 240\beta_1^3 + 1164\beta_1^2 \\ & + (112\beta_2^3 + (-352\beta + 196)\beta_2^2 + (320\beta_1^2 - 268\beta_1 + \frac{49}{2})\beta_2 - 64\beta_1^3 - 96\beta_1^2 \\ & + (40\beta_2^3 + (-80\beta_1 - 42)\beta_2^2 + (56\beta_1^2 + 92\beta_1 - 50)\beta_2 - 16\beta^3 - 18\beta_1^2 \\ & - 21\beta_1 + \frac{257}{8})x^4 \\ & + (-8\beta_2^3 + (24\beta_1 - 32)\beta_2^2 + (-24\beta_1^2 + 52\beta_1 - 3)\beta_2 + 8\beta_1^3 - 24\beta_1^2 + 9\beta_1 - 4)x^3 \\ & + (6\beta_2^2 + (-12\beta_1 + \frac{19}{2})\beta_2 + 6\beta^2 - 9\beta_1 + \frac{19}{8})x^2 + \left(\frac{3\beta_1}{2} - \frac{3\beta_2}{2} - 1\right)x. \end{aligned}$$

The remaining coefficients follow the same structural pattern and are therefore omitted for brevity. The complete symbolic expressions can be generated using a computer algebra system such as Maple.

References

1. Kelley, C.T. *Iterative Methods for Linear and Nonlinear Equations*; SIAM: Philadelphia, PA, USA, 1995. [\[CrossRef\]](#)
2. Costabile, F.; Gualtieri, M.I.; Luceri, R. A new iterative method for the computation of the solutions of nonlinear equations. *Numer. Algorithms* **2001**, *28*, 87–100. [\[CrossRef\]](#)
3. Ran, C.; Xu, X.; Hou, C.; Zhang, X. Numerical approximation of fourth-order fractional diffusion-wave systems using finite difference and discontinuous Galerkin method. *Netw. Heterog. Media* **2025**, *20*, 1346–1366. [\[CrossRef\]](#)
4. Zhang, X.; Wang, H.; Luo, Z.; Wei, L. A high-accuracy compact finite difference scheme for time-fractional diffusion equations. *Rev. Unión Mat. Argent.* **2025**, *24*, 1952–1968. [\[CrossRef\]](#)
5. Baleanu, D. *Fractional Calculus: Models and Numerical Methods*; World Scientific: Singapore, 2012; Volume 3. [\[CrossRef\]](#)
6. Privman, V. (Ed.) *Finite Size Scaling and Numerical Simulation of Statistical Systems*; World Scientific: Singapore, 1990. [\[CrossRef\]](#)
7. Peng, B.; Zhang, Y. A Fractional Calculus Framework for Open Quantum Dynamics: From Liouville to Lindblad to Memory Kernels. *arXiv* **2025**, arXiv:2511.13038. [\[CrossRef\]](#) [\[PubMed\]](#)
8. Siddi Moreau, G.; Pisani, L.; Profir, M.; Podda, C.; Leoni, L.; Cao, G. Quantum artificial intelligence scalability in the NISQ era: Pathways to Quantum Utility. *Adv. Quantum Technol.* **2025**, *8*, 2400716. [\[CrossRef\]](#)
9. Varshney, V.; Kingston, S.L.; Srinivasan, S.; Kumarasamy, S. Hidden attractors in fractional-order discrete maps. *Eur. Phys. J. B* **2024**, *97*, 161. [\[CrossRef\]](#)
10. Wang, X.; Zhang, J. Dynamic analysis and circuit design of tunable multi-vortex chaotic systems based on memristors. *Nonlinear Dyn.* **2024**, *112*, 14415–14440. [\[CrossRef\]](#)
11. Ali, N.; Waseem, M.; Safdar, M.; Akgül, A.; Tolasa, F.T. Iterative solutions for nonlinear equations via fractional derivatives: Adaptations and advances. *Appl. Math. Sci. Eng.* **2024**, *32*, 2333816. [\[CrossRef\]](#)
12. Sana, G.; Mohammed, P.O.; Shin, D.Y.; Noor, M.A.; Oudat, M.S. On iterative methods for solving nonlinear equations in quantum calculus. *Fractal Fract.* **2021**, *5*, 60. [\[CrossRef\]](#)
13. Nonlaopon, K.; Khan, A.G.; Ameen, F.; Awan, M.U.; Cesarano, C. Some new quantum numerical techniques for solving nonlinear equations. *Symmetry* **2022**, *14*, 1829. [\[CrossRef\]](#)
14. Herdem, S.; Erkövan, S. On the construction of iterative methods in (p, q) -calculus. *Numer. Algorithms* **2026**, 1–25. [\[CrossRef\]](#)
15. Shams, M.; Carpentieri, B. Q-analogues of parallel numerical scheme based on neural networks and their engineering applications. *Appl. Sci.* **2024**, *14*, 1540. [\[CrossRef\]](#)

16. Gulshan, G.; Budak, H.; Hussain, R.; Sadiq, A. Generalization of the bisection method and its applications in nonlinear equations. *Adv. Contin. Discret. Model.* **2023**, *2023*, 18. [CrossRef]
17. Qureshi, S.; Soomro, A.; Argyros, I.K.; Gdawiec, K.; Akgül, A.; Alquran, M. Use of fractional calculus to avoid divergence in Newton-like solver for solving one-dimensional nonlinear polynomial-based models. *Commun. Nonlinear Sci. Numer. Simul.* **2025**, *143*, 108631. [CrossRef]
18. Tay, T.T.; Mareels, I.; Moore, J.B. Adaptive-Q application to nonlinear systems. In *High Performance Control*; Birkhäuser: Boston, MA, USA, 1998; pp. 205–240. [CrossRef]
19. Candelario, G.; Cordero, A.; Torregrosa, J.R.; Vassileva, M.P. An optimal and low computational cost fractional Newton-type method for solving nonlinear equations. *Appl. Math. Lett.* **2022**, *124*, 107650. [CrossRef]
20. Shams, M.; Kausar, N.; Carpentieri, B. A class of high-order fractional parallel iterative methods for nonlinear engineering problems: Convergence, stability, and neural network-based acceleration. *Chaos Solitons Fractals* **2025**, *199*, 116646. [CrossRef]
21. Akram, S.; Junjua, M.-U.-D.; Akram, F.; Khalid, L. Iterative Techniques for Nonlinear Equations: Addressing Multiple Roots with Unknown Multiplicity. In *Polynomials-Exploring Fundamental Mathematical Expressions*; IntechOpen: London, UK, 2025; p. 33. [CrossRef]
22. Batiha, I.M.; Al-Khawaldeh, H.O.; Jebril, I.H.; Anakira, N.; Bataihah, A.; Momani, S. A Novel Numerical Approach for Handling Fractional Quantum Initial-Value Problems. *Int. J. Fuzzy Log. Intell. Syst.* **2025**, *25*, 416–430. [CrossRef]
23. Kac, V.G.; Cheung, P. *Quantum Calculus*; Springer: New York, NY, USA, 2002. [CrossRef]
24. Elsayir, H.A.; Altoum, S.H. Formulation of the Schrödinger equation using the Jackson q -derivative. *J. Comput. Anal. Appl.* **2026**, *35*, 1189–1196. Available online: <https://eudoxuspress.com/index.php/pub/article/view/5040> (accessed on 10 April 2026).
25. Jackson, F.H. q -Difference equations. *Am. J. Math.* **1910**, *32*, 305–314. [CrossRef]
26. Njionou Sadjang, P. On the fundamental theorem of (p, q) -calculus and some (p, q) -Taylor formulas. *Results Math.* **2018**, *73*, 39. [CrossRef]
27. Askey, R. The q -gamma and q -beta functions. *Appl. Anal.* **1978**, *8*, 125–141. [CrossRef]
28. Kalla, S.; Yadav, R.; Purohit, S. On the Riemann-Liouville fractional q -integral operator involving a basic analogue of Fox H -function. *Fract. Calc. Appl. Anal.* **2005**, *8*, 313–322. Available online: <http://hdl.handle.net/10525/1260> (accessed on 10 April 2026).
29. Shaimardan, S.; Tokmagambetov, N.S. On the solutions of some fractional q -differential equations with the Riemann-Liouville fractional q -derivative. *Bull. Karaganda Univ. Math. Ser.* **2021**, *4*, 130–141. [CrossRef]
30. Deppman, A.; Megías, E.; Pasechnik, R. Fractal derivatives, fractional derivatives and q -deformed Calculus. *Entropy* **2023**, *25*, 1008. [CrossRef]
31. Allouch, N.; Batiha, I.M.; Jebril, I.H.; Hamani, S.; Al-Khateeb, A.; Momani, S. A new fractional approach for the higher-order q -Taylor method. *Image Anal. Stereol.* **2024**, *43*, 249–257. [CrossRef]
32. Beardon, A.F. *Iteration of Rational Functions*; Springer: New York, NY, USA, 2000. Available online: <https://link.springer.com/book/9780387951515> (accessed on 10 April 2026).
33. Argyros, I.K.; Magreñán, Á.A. *Iterative Methods and Their Dynamics with Applications—A Contemporary Study*; CRC Press: Boca Raton, FL, USA, 2017. [CrossRef]
34. Fogler, H.S. *Elements of Chemical Reaction Engineering*, 3rd ed.; Prentice Hall: Upper Saddle River, NJ, USA, 1999.
35. Bergman, R.N.; Phillips, L.S.; Cobelli, C. Physiologic evaluation of factors controlling glucose tolerance in man: Measurement of insulin sensitivity and beta-cell glucose sensitivity from the response to intravenous glucose. *J. Clin. Investig.* **1981**, *68*, 1456–1467. [CrossRef]
36. Chua, L.O. Nonlinear circuit foundations for nanodevices. I. The four-element torus. *Proc. IEEE* **2004**, *91*, 1830–1859. [CrossRef]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.