Eur. J. Math. Anal. 6 (2026) 15
doi: 10.28924/ada/ma.6.15

A Two-Step Conformable Fractional Iterative Method with Stability and Applications to Nonlinear Problems

Nishant Kumar, Toshan Kumar Shriwas, J. P. Jaiswal*

Department of Mathematics, Guru Ghasidas Vishwavidyalaya (A Central University), Bilaspur (C.G.), India

nishantyd001@gmail.com, toshan.shriwas786@gmail.com, asstprofjpmanit@gmail.com

* *Correspondence: asstprofjpmanit@gmail.com*

Abstract. In this paper, a two-step iterative method based on conformable fractional derivatives is proposed for solving nonlinear equations. The convergence analysis confirms that the method achieves fourth-order convergence. Numerical experiments on real-world problems reveal that the method significantly reduces the number of iterations compared to classical schemes. The method is also tested on several important models, including the Van der Waals equation, blood flow dynamics, population growth laws, chemical engineering models, and Planck's radiation law, demonstrating its wide applicability. Furthermore, the basin of attraction analysis shows improved stability and efficiency, highlighting the method's potential as a reliable alternative for nonlinear problem-solving.

1. Introduction

The conceptual foundation of fractional calculus can be traced back to 1695, when Leibniz and L'Hôpital initiated a discussion on the possibility of defining derivatives of arbitrary, non-integer order—such as the intriguing case of a derivative of order one-half. Since then, researchers have introduced a variety of definitions for fractional derivatives that retain key features of traditional calculus. These formulations have proven particularly useful for modeling intricate behaviors in real-world systems, making them a valuable asset in numerous scientific and engineering applications [3, 10].

Several recent contributions in the literature have introduced Newton-type iterative algorithms incorporating fractional derivatives such as the Caputo and Riemann–Liouville definitions [1, 4, 9, 16]. These methods are designed to approximate solutions $\bar{x} \in \mathbb{R}$ of nonlinear equations, where the function $g : I \subset \mathbb{R} \to \mathbb{R}$ remains continuous within its domain. Nonetheless, these fractional schemes often struggle to maintain the theoretically predicted order of convergence when applied in practice. To address this issue, enhanced Newton-type iterations

Received: 19 Apr 2026.

2020 *Mathematics Subject Classification.* 65H05, 26A33, 41A25.

Key words and phrases. Conformable fractional derivative, Nonlinear equations, Stability, Multipoint methods.

based on the conformable derivative were proposed in [5,6], both in scalar and vector formulations. These revised techniques not only retain the expected convergence rate but also exhibit strong alignment between theoretical analysis and numerical performance.

We begin by presenting some fundamental aspects of conformable fractional calculus. Let $g : [a, \infty) \to \mathbb{R}$ be a function defined on the real line, and consider a point $x > a$, where $a \in \mathbb{R}$. The left conformable derivative [13,19] of order $\alpha \in (0, 1]$ starting from a is given by:

$$(T^a_\alpha g)(x) = \lim_{\epsilon \to 0} \frac{g(x + \epsilon(x - a)^{1-\alpha}) - g(x)}{\epsilon}. \tag{1.1}$$

If this limit exists, the function g is said to be conformably differentiable of order α. In the case where g is differentiable, this derivative reduces to a simpler form:

$$(T^a_\alpha g)(x) = (x - a)^{1-\alpha} g'(x). \tag{1.2}$$

Moreover, if g is conformably differentiable on an interval (a, b) for some $b \in \mathbb{R}$, then the value of the conformable derivative at the left endpoint a is obtained via the right-hand limit $(T^a_\alpha g)(a) = \lim_{x \to a^+} (T^a_\alpha g)(x)$.

One of the notable features of this definition is that it retains properties analogous to those of the classical derivative. For instance, the conformable derivative of a constant function is zero, i.e., $T^a_\alpha C = 0$ for any constant C. In contrast to other popular fractional derivatives such as the Riemann–Liouville and Caputo derivatives, the conformable approach does not depend on special functions like the Gamma or Mittag-Leffler functions. This simplicity makes it particularly attractive for both theoretical and computational applications.

The following theorem presents a generalized Taylor series representation for a function $g(x)$, wherein the conformable derivatives are computed starting from a reference point a, which is distinct from the location a_1 at which the series is evaluated. .

Theorem 1.1. *(Theorem 4.1, [18]). Assume that the function $g(x)$ is infinitely many times differentiable in the conformable sense of order α where $0 < \alpha \leq 1$, within a neighborhood of the point a_1. Suppose that these conformable derivatives are taken with respect to a fixed point a. Then, the function $g(x)$ admits the following conformable Taylor-type expansion centered at a_1:*

$$g(x) = g(a_1) + \frac{(T^a_\alpha g)(a_1)\delta_1}{\alpha} + \frac{(T^a_\alpha g)^2(a_1)\delta_2}{2\alpha^2} + R_2(x, a_1, a), \tag{1.3}$$

where $T^a_\alpha g$ denotes the conformable derivative of order α with respect to a, and $\delta_1 = H^\alpha - L^\alpha$, $\quad \delta_2 = H^{2\alpha} - L^{2\alpha} - 2L^\alpha \delta_1, \cdots$, with $H = x - a$ and $L = a_1 - a$.

Furthermore, for higher-order terms, it can be shown that $\delta_2 = \delta_1^2$, $\delta_3 = \delta_1^3$, and so on, suggesting a pattern that simplifies the expansion. Hence, the expression for $g(x)$ can be compactly rewritten as:$\leq$

$$g(x) = g(a_1) + \frac{(T^a_\alpha g)(a_1)\delta_1}{\alpha} + \frac{(T^a_\alpha g)^2(a_1)\delta_1^2}{2\alpha^2} + R_2(x, a_1, a), \tag{1.4}$$

where $R_2(x, a_1, a)$ represents the second-order remainder term.

In this work, we propose a novel two-step fractional iterative method designed for solving nonlinear equations, accompanied by a detailed convergence analysis. The paper is structured as follows: Section 2 introduces the construction of the new two-point iterative scheme, formulated using the conformable fractional derivative. The proposed method achieves a fourth-order rate of convergence. In Section 3, numerical experiments inspired by real-world scenarios are carried out to evaluate the effectiveness and efficiency of the method. Section 4 presents a comparative stability analysis between the newly developed approach and several existing techniques. Finally, Section 5 concludes the study by summarizing key findings and potential directions for future research.

2. Development of a Two-Step Fractional Iterative Method

In this section, we introduce a new two-step fractional iterative method by extending an existing two-step classical iterative approach. Specifically, we replace the classical derivative with the conformable fractional derivative to construct a generalized fractional scheme, inspired by the method proposed by Shengfeng Li [15]. The original two-step iterative method is given by:

$$\begin{aligned} y_n &= x_n - \frac{g(x_n)}{g'(x_n)}, \\ x_{n+1} &= y_n - \left(\frac{g(x_n)-g(y_n)}{g(x_n)-2g(y_n)}\right)\frac{g(x_n)}{g'(x_n)}. \end{aligned} \tag{2.1}$$

To extend this method within a fractional framework, we replace the classical derivative $g'(x_n)$ with the conformable derivative $(T^a_\alpha g)(x_n)$. Consequently, the proposed two-step fractional iterative method, denoted by NTJ, is formulated as follows:

$$\begin{aligned} y_n &= a + \left((x_n-a)^\alpha - \alpha\frac{g(x_n)}{(T^a_\alpha g)(x_n)}\right)^{\frac{1}{\alpha}}, \\ x_{n+1} &= a + \left((x_n-a)^\alpha - \alpha\left(\frac{g(x_n)-g(y_n)}{g(x_n)-2g(y_n)}\right)\frac{g(x_n)}{(T^a_\alpha g)(x_n)}\right)^{\frac{1}{\alpha}}. \end{aligned} \tag{2.2}$$

This new approach effectively generalizes the classical method into the fractional setting, aiming to achieve higher flexibility and improved convergence properties.

We establish the following result regarding the convergence theorem for the fractional iterative technique given in (2.2).

Theorem 2.1. *Let $g : I \to \mathbb{R}$ be continuous on an interval I containing a simple zero $\bar{x}$. Assume the conformable derivative $(T^a_\alpha g)(x)$, with order $\alpha \in (0,1]$, exists, is continuous, and non-vanishing at $\bar{x}$. If x_0 is chosen close enough to $\bar{x}$, the two-step fractional scheme (NTJ) achieves at least order 4 of convergence, and the associated error equation is:*

$$e_{n+1} = \left(\frac{\alpha^2}{2}(\alpha-1)(\bar{x}-a)^{2\alpha-3}(3-5d_2^2) + \alpha^3(\bar{x}-a)^{3\alpha-3}(3d_2^2-d_2d_3)\right)e_n^4 + O(e_n^5), \tag{2.3}$$

where the coefficients d_k for $k = 2, 3, 4, \dots$ are defined as

$$d_k = \frac{1}{k!\alpha^k} \frac{(T^a_\alpha g)^{(k)}(\bar{x})}{(T^a_\alpha g)(\bar{x})}.$$

Proof. Applying the Taylor series expansion (1.4) (see ref. [18]) for $g(x_n)$ around $\bar{x}$, and setting $x_n = e_n + \bar{x}$, we obtain:

$$\begin{aligned} g(x_n) = (T^a_\alpha g)(\bar{x}) \Big[& (e_n + \bar{x} - a)^\alpha - (\bar{x} - a)^\alpha + d_2 \left((e_n + \bar{x} - a)^\alpha - (\bar{x} - a)^\alpha \right)^2 \\ & + d_3 \left((e_n + \bar{x} - a)^\alpha - (\bar{x} - a)^\alpha \right)^3 + d_4 \left((e_n + \bar{x} - a)^\alpha - (\bar{x} - a)^\alpha \right)^4 \Big] + O(e_n^5), \end{aligned}$$

Next, recall the generalized binomial expansion [5] for fractional exponents:

$$(x+y)^r = \sum_{k=0}^{\infty} \binom{r}{k} x^{r-k} y^k, \quad \text{for } k \in \{0\} \cup \mathbb{N},$$

where the binomial coefficients are generalized by

$$\binom{r}{k} = \frac{\Gamma(r+1)}{k!\,\Gamma(r-k+1)}.$$

Using this, we can further expand $g(x_n)$ as:

$$\begin{aligned} g(x_n) = \frac{(T^a_\alpha g)(\bar{x})}{\alpha} \Big[& \alpha(\bar{x} - a)^{\alpha-1} e_n + \left(\frac{\alpha}{2}(\alpha - 1)(\bar{x} - a)^{\alpha-2} + \alpha^2 (\bar{x} - a)^{2\alpha-2} d_2 \right) e_n^2 \\ & + \left(\frac{\alpha}{6}(\alpha-1)(\alpha-2)(\bar{x} - a)^{\alpha-3} + \alpha^2(\alpha-1)(\bar{x} - a)^{2\alpha-3} d_2 + \alpha^3 (\bar{x} - a)^{3\alpha-3} d_3 \right) e_n^3 \\ & + \Big(\frac{\alpha}{24}(\alpha-1)(\alpha-2)(\alpha-3)(\bar{x} - a)^{\alpha-4} + \frac{\alpha^2}{4}(\alpha-1)^2(\bar{x} - a)^{3\alpha-4} d_2 + \frac{\alpha^2}{3}(\alpha-1) \times \\ & (\alpha-2)(\bar{x} - a)^{2\alpha-4} d_2 + \frac{3}{2}\alpha^3(\alpha-1)(\bar{x} - a)^{3\alpha-4} d_3 + \alpha^4 (\bar{x} - a)^{4\alpha-4} d_4 \Big) e_n^4 + \cdots \Big]. \end{aligned} \tag{2.4}$$

Knowing that the conformable derivative satisfies $(T^a_\alpha g)(x) = (x-a)^{1-\alpha} g'(x)$, and by utilizing the generalized binomial expansion again, the conformable derivative of $g(x_n)$ can be expanded as

$$\begin{aligned} (T^a_\alpha g)(x_n) = (T^a_\alpha g)(\bar{x}) \Big[& 1 + \left(2\alpha(\bar{x} - a)^{\alpha-1} d_2 \right) e_n + \left(\alpha(\alpha-1)(\bar{x} - a)^{\alpha-2} d_2 + 3\alpha^2 (\bar{x} - a)^{2\alpha-2} d_3 \right) e_n^2 \\ & + \left(\frac{\alpha}{3}(\alpha-1)(\alpha-2)(\bar{x} - a)^{\alpha-3} d_2 + 3\alpha^2(\alpha-1)(\bar{x} - a)^{2\alpha-3} d_3 + 4\alpha^3 (\bar{x} - a)^{3\alpha-3} d_4 \right) e_n^3 \Big] \\ & + O(e_n^4). \end{aligned} \tag{2.5}$$

After performing some simplifications, we obtain

$$\begin{aligned}\frac{g(x_n)}{(T^a_\alpha g)(x_n)} &= \frac{1}{\alpha}\Bigg[\alpha(\bar{x}-\alpha)^{\alpha-1}e_n + \left\{\frac{\alpha}{2}(\alpha-1)(\bar{x}-\alpha)^{\alpha-2} - \alpha^2(\bar{x}-\alpha)^{\alpha-2}d_2\right\}e_n^2 \\ &+ \left\{\frac{\alpha}{6}(\alpha-1)(\alpha-2)(\bar{x}-\alpha)^{\alpha-3} + \alpha^3(\bar{x}-\alpha)^{\alpha-3}(2d_2^2-2d_3) - \alpha^2(\alpha-1)(\bar{x}-\alpha)^{2\alpha-3}d_2\right\}e_n^3 \\ &+ \Bigg\{\frac{\alpha}{24}(\alpha-1)(\alpha-2)(\alpha-3)(\bar{x}-\alpha)^{\alpha-4} + \frac{\alpha^2}{4}(\alpha-1)^2(\bar{x}-\alpha)^{2\alpha-4}d_2 \\ &-\alpha^2(\alpha-1)(\bar{x}-\alpha)^{2\alpha-4}d_2\left(\frac{5}{6}\alpha-\frac{7}{6}\right) + \alpha^3(\alpha-1)(\bar{x}-\alpha)^{3\alpha-4}(3d_2^2-3d_3) \\ &+\alpha^4(\bar{x}-\alpha)^{4\alpha-4}\left(-4d_2^2+7d_2d_3-3d_4\right)\Bigg\}e_n^4 + \cdots\Bigg],\end{aligned}$$

consequently, we get

$$\begin{aligned}(x_n-a)^\alpha - \alpha\frac{g(x_n)}{(T^a_\alpha g)(x_n)} &= (\bar{x}-\alpha)^\alpha + \alpha^2(\bar{x}-\alpha)^{2\alpha-2}d_2e_n^2 \\ &+ \left\{\alpha^2(\alpha-1)(\bar{x}-\alpha)^{2\alpha-3}d_2 - \alpha^3(\bar{x}-\alpha)^{3\alpha-3}(2d_2^2-2d_3)\right\}e_n^3 \\ &- \Bigg\{\frac{\alpha^2}{4}(\alpha-1)^2(\bar{x}-\alpha)^{2\alpha-2}d_2 - \alpha^2(\alpha-1)(\bar{x}-\alpha)^{2\alpha-4}d_2\left(\frac{5}{6}\alpha-\frac{7}{6}\right) \\ &+ \alpha^3(\alpha-1)(\bar{x}-\alpha)^{3\alpha-4}(3d_2^2-3d_3) + \alpha^4(\bar{x}-\alpha)^{4\alpha-4}(7d_2d_3-4d_2^3-3d_4)\Bigg\}e_n^4 + \cdots\end{aligned}$$

Applying the generalized Binomial theorem once more, we obtain
$\left[(x_n-a)^\alpha - \alpha\frac{g(x_n)}{(T^a_\alpha g)(x_n)}\right]^{\frac{1}{\alpha}}$

$$\begin{aligned}&= (\bar{x}-a) + \Bigg[\alpha(\bar{x}-a)^{\alpha-1}d_2e_n^2 + \left\{\alpha(\alpha-1)(\bar{x}-a)^{\alpha-2}d_2 - \alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-2}(2d_2^2-2d_3)\right\}e_n^3 \\ &- \Bigg\{\frac{\alpha}{4}(\alpha-1)^2(\bar{x}-a)^{\alpha-2} - \alpha(\alpha-1)(\bar{x}-a)^{\alpha-3}d_2(\frac{5}{6}\alpha-\frac{7}{6}) \quad + \alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-3}(3d_2^2-3d_3) \\ &+ \alpha^3(\bar{x}-a)^{3\alpha-3}(7d_3-4d_2^3-3d_4)\Bigg\}e_n^4 + \cdots\Bigg] \quad + \frac{1}{2}\left(\frac{1}{\alpha}-1\right)\left[\alpha^3(\bar{x}-a)^{2\alpha-3}d_2^2\,e_n^4 + O(e_n^5)\right] + \cdots \end{aligned} \tag{2.6}$$

Now

$$y_n = a + \left[(x_n-a)^\alpha - \alpha\frac{g(x_n)}{(T^a_\alpha g)(x_n)}\right]^{1/\alpha}. \tag{2.7}$$

Therefore, by applying equation (2.6) into the equation (2.7)

$$\begin{aligned}e_{n,y} + \bar{x} = a + \bar{x} - a &+ \alpha(\bar{x}-a)^{\alpha-1}d_2e_n^2 \\ &+ \left\{\alpha(\alpha-1)(\bar{x}-a)^{\alpha-2}d_2 - 2\alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-2}(d_2^2-d_3)\right\}e_n^3 \\ &- \Bigg\{\frac{\alpha}{4}(\alpha-1)^2(\bar{x}-a)^{\alpha-3}d_2 - \alpha(\alpha-1)(\bar{x}-a)^{\alpha-3}d_2(\frac{5}{6}\alpha-\frac{7}{6}) \\ &+ \alpha^2(\bar{x}-a)^{2\alpha-3}\left(3d_2^2-3d_3\right) + \alpha^3(\bar{x}-a)^{3\alpha-3}(7d_3-4d_2^3-3d_4) \\ &- \frac{1}{2}\left(\frac{1}{\alpha}-1\right)\alpha^3(\bar{x}-a)^{2\alpha-3}d_2^2\Bigg\}e_n^4 + \cdots,\end{aligned} \tag{2.8}$$

$$= \alpha(\bar{x}-a)^{\alpha-1}d_2e_n^2 \quad + \left\{\alpha(\alpha-1)(\bar{x}-a)^{\alpha-2}d_2 - 2\alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-2}(d_2^2-d_3)\right\}e_n^3$$
$$- \left\{\frac{\alpha}{4}(\alpha-1)^2(\bar{x}-a)^{\alpha-3}d_2 - \alpha(\alpha-1)(\bar{x}-a)^{\alpha-3}d_2\left(\frac{5}{6}\alpha-\frac{7}{6}\right)\right.$$
$$+ \alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-3}(3d_2^2-3d_3) \quad + \alpha^3(\bar{x}-a)^{3\alpha-3}(7d_2d_3-4d_2^3-3d_4)$$
$$\left.- \frac{1}{2}\left(\frac{1}{\alpha}-1\right)\alpha^3(\bar{x}-a)^{2\alpha-3}d_2^2\right\}e_n^4 + \cdots. \tag{2.9}$$

Now, we have

$$g(y_n) = \frac{(T_\alpha^a g)(\bar{x})}{\alpha}\left[\alpha(\bar{x}-a)^{\alpha-1}e_{n,y} + \left\{\frac{\alpha}{2}(\alpha-1)(\bar{x}-a)^{\alpha-2} + \alpha^2(\bar{x}-a)^{2\alpha-2}d_2\right\}e_{n,y}^2 + \cdots\right]. \tag{2.10}$$

Substituting equation (2.9) into expression (2.10) and performing some simplifications, we obtain

$$g(y_n) = \frac{(T_\alpha^a g)(\bar{x})}{\alpha}\left[\alpha^2(\bar{x}-a)^{2\alpha-2}d_2e_n^2 + \left\{\alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-3}d_2 - 2\alpha^3(\bar{x}-a)^{3\alpha-3}(d_2^2-d_3)\right\}e_n^3\right.$$
$$- \left\{\alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-4}d_2\left(-\frac{7}{12}\alpha+\frac{11}{12}\right) + \alpha^3(\alpha-1)(\bar{x}-a)^{3\alpha-4}\left(2d_2^2-3d_3\right)\right.$$
$$\left.\left.+ \alpha^4(\bar{x}-a)^{4\alpha-4}\left(7d_2d_3-3d_2^3-3d_4\right)\right\}e_n^4 + O(e_n^5)\right].$$

Now,

$$g(x_n) - g(y_n) = \frac{(T_\alpha^a g)(\bar{x})}{\alpha}\left[\alpha(\bar{x}-a)^{\alpha-1}e_n + \frac{\alpha}{2}(\alpha-1)(\bar{x}-a)^{\alpha-2}e_n^2\right.$$
$$+ \left\{\frac{\alpha}{6}(\alpha-1)(\alpha-2)(\bar{x}-a)^{\alpha-3} + \alpha^3(\bar{x}-a)^{3\alpha-3}(2d_2^2-d_3)\right\}e_n^3$$
$$+ \left\{\frac{\alpha}{24}(\alpha-1)(\alpha-2)(\alpha-3)(\bar{x}-a)^{\alpha-4} + \alpha^3(\alpha-1)(\bar{x}-a)^{3\alpha-4}\left(-\frac{9}{2}d_3+4d_2^2\right)\right.$$
$$\left.\left.+ \alpha^4(\bar{x}-a)^{4\alpha-4}\left(-2d_4+7d_2d_3-3d_2^3\right)\right\}e_n^4 + O(e_k^5)\right], \tag{2.11}$$

and

$$g(x_n) - 2g(y_n) = (T_\alpha^a g)(\bar{x})(\bar{x}-a)^{\alpha-1}e_n\left[1 + \left\{\frac{1}{2}(\alpha-1)(\bar{x}-a)^{-1} - \alpha(\bar{x}-a)^{\alpha-1}d_2\right\}e_n\right.$$
$$+ \left\{\frac{1}{6}(\alpha-1)(\alpha-2)(\bar{x}-a)^{-2} - \alpha(\alpha-1)(\bar{x}-a)^{\alpha-2}d_2 + \alpha^2(\bar{x}-a)^{2\alpha-2}(4d_2^2-3d_3)\right\}e_n^2$$
$$+ \left\{\left(\frac{1}{24}(\alpha-1)(\alpha-2)(\alpha-3)(\bar{x}-a)^{-3} + \alpha(\alpha-1)(\bar{x}-a)^{\alpha-3}d_2\left(-\frac{7}{12}\alpha+\frac{11}{12}\right)\right.\right.$$
$$\left.\left.+ \alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-3}\left(8d_2^2-\frac{9}{2}d_3\right) + \ \alpha^3(\bar{x}-a)^{3\alpha-3}(14d_2d_3-6d_2^3-5d_4)\right)\right\}e_n^3$$
$$\left.+ O(e_n^4)\right]. \tag{2.12}$$

From (2.11) and (2.12), we arrive at the following expression

$$\frac{g(x_n)-g(y_n)}{g(x_n)-2g(y_n)} = 1 + \alpha(\bar{x}-a)^{\alpha-1}d_2e_n + \left\{\frac{1}{2}\alpha(\alpha-1)(\bar{x}-a)^{\alpha-2}d_2 + \alpha^2(\bar{x}-a)^{2\alpha-2}(2d_3-d_2^2)\right\}e_n^2$$
$$+ \left\{\frac{\alpha}{6}(\alpha-1)(\alpha-2)(\bar{x}-a)^{\alpha-3}d_2 + \alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-3}\left(-2d_2^2+2d_3\right)\right.$$
$$\left.+ \alpha^3(\bar{x}-a)^{3\alpha-3}(-2d_2d_3-2d_2^3+3d_4)\right\}e_n^3 + O(e_n^4).$$

Now,

$$\frac{g(x_n)-g(y_n)}{g(x_n)-2g(y_n)}\frac{g(x_n)}{(T_\alpha^a g)(x_n)}$$
$$=\frac{1}{\alpha}\Bigg[\alpha(\bar{x}-a)^{\alpha-1}e_n+\frac{1}{2}\alpha(\alpha-1)(\bar{x}-a)^{\alpha-2}e_n^2+\frac{\alpha}{6}(\alpha-1)(\alpha-2)(\bar{x}-a)^{\alpha-3}e_n^3$$
$$+\Bigg\{\frac{\alpha}{24}(\alpha-1)(\alpha-2)(\alpha-3)(\bar{x}-a)^{\alpha-4}+\alpha^3(\alpha-1)(\bar{x}-a)^{3\alpha-4}\left(-d_2^2\right)$$
$$+\alpha^4(\alpha-1)(\bar{x}-a)^{4\alpha-4}\left(d_2d_3-3d_2^3\right)\Bigg\}e_n^4+O(e_k^5)\Bigg].$$

Thus, by applying the generalized Binomial theorem, we obtain the result

$$\left[(x_n-a)^\alpha-\alpha\frac{g(x_n)-g(y_n)}{g(x_n)-2g(y_n)}\frac{g(x_n)}{(T_\alpha^a g)(x_n)}\right]$$
$$=(\bar{x}-a)^\alpha+\left\{\alpha^3(\alpha-1)(\bar{x}-a)^{3\alpha-4}d_2^2+\alpha^4(\bar{x}-a)^{4\alpha-4}(3d_2^3-d_3d_2)\right\}e_n^4+O(e_n^5).$$

Next,

$$\left[(x_n-a)^\alpha-\alpha\frac{g(x_n)-g(y_n)}{g(x_n)-2g(y_n)}\frac{g(x_n)}{(T_\alpha^a g)(x_n)}\right]^{\frac{1}{\alpha}}$$
$$=(\bar{x}-a)+\frac{1}{\alpha}\left[\left\{\alpha^3(\alpha-1)(\bar{x}-a)^{3\alpha-4}d_2^2+\alpha^4(\bar{x}-a)^{4\alpha-4}(3d_2^3-d_2d_3)\right\}e_n^4\right]((\bar{x}-a)^\alpha)^{\frac{1}{\alpha}-1}$$
$$=\bar{x}-a+\left\{\alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-3}d_2^2+\alpha^3(\bar{x}-a)^{3\alpha-3}(3d_2^3-d_2d_3)\right\}e_n^4+O(e_n^5). \tag{2.13}$$

Employing the substitution $x_{n+1}=e_{n+1}+\bar{x}$ in equation (2.13), the following expression is obtained.

$$e_{n+1}+\bar{x}=a+\bar{x}-a+\left\{\frac{\alpha^2}{2}(\alpha-1)(\bar{x}-a)^{2\alpha-3}d_2^2+\alpha^3(\bar{x}-a)^{3\alpha-3}(3d_2^3-d_2d_3)\right\}e_n^4+O(e_n^5),$$

or,

$$e_{n+1}=\left\{\alpha^2(\alpha-1)(\bar{x}-a)^{2\alpha-3}d_2^2+\alpha^3(\bar{x}-a)^{3\alpha-3}(3d_2^3-d_2d_3)\right\}e_n^4+O(e_n^5).$$

It follows that the method presented in (2.2) achieves a convergence order of four. □

3. Numerical Results and Comparison

In this numerical investigation, we present a detailed comparison of various fractional iterative methods developed to solve nonlinear equations arising from practical, real-world applications. The primary objective is to evaluate the performance of our recently proposed fractional iterative method (NTJ) in comparison with existing approaches such as TeCo, CKeCo, and CeCo from [7], and to demonstrate its effectiveness across a diverse set of problems. Accordingly, several application-oriented examples are solved numerically using different iterative schemes, and the corresponding results are analyzed in this section.

The numerical experiments were carried out using **MATHEMATICA 8** with multiple precision of 200 digits. Each method was executed with a maximum of 500 iterations, and the stopping criterion was defined as either $|g(x_{n+1})|<10^{-8}$ or $|x_{n+1}-x_n|<10^{-8}$. For each test function, we choose $a=-10$ to ensure that $a<x_n$ for all n. Additionally, we consider

equispaced values of $\alpha \in (0,1]$. The outcomes of these experiments are summarized in Tables 1 through 5, which report the following performance metrics for each method:

- The absolute residual error $|g(x_n)|$,
- The absolute error in approximating the root $|x_{n+1} - x_n|$,
- The number of iterations required to satisfy the stopping criterion,
- The number of function evaluations (NFE) per iteration,
- The computational order of convergence (COC) [2], given by:

$$\mathrm{COC} \approx \frac{\log\left|\frac{x_{n+1}-x_n}{x_n-x_{n-1}}\right|}{\log\left|\frac{x_n-x_{n-1}}{x_{n-1}-x_{n-2}}\right|}.$$

This metric serves as an indicator of how rapidly the iterative sequence converges to the true solution.

In 2023, Candelario *et al.* [7] introduced several two-step fractional iterative methods based on the use of the conformable derivative. These techniques are described as follows:

TeCo is defined as

$$\begin{aligned} y_n &= a + \left((x_n - a)^\alpha - \alpha \frac{g(x_n)}{(T_\alpha^a g)(x_n)}\right)^{\frac{1}{\alpha}}, \\ x_{n+1} &= a + \left((y_n - a)^\alpha - \alpha \frac{g(y_n)}{(T_\alpha^a g)(x_n)}\right)^{\frac{1}{\alpha}}. \end{aligned} \tag{3.1}$$

CKeCo is defined as

$$\begin{aligned} y_n &= a + \left((x_n - a)^\alpha - \alpha \frac{g(x_n)}{(T_\alpha^a g)(x_n)}\right)^{\frac{1}{\alpha}}, \\ x_{n+1} &= a + \left((x_n - a)^\alpha - \frac{\alpha}{2}\left[3 - \frac{(T_\alpha^a g)(y_n)}{(T_\alpha^a g)(x_n)}\right] \frac{g(x_n)}{(T_\alpha^a g)(x_n)}\right)^{\frac{1}{\alpha}}. \end{aligned} \tag{3.2}$$

CeCo is defined as

$$\begin{aligned} y_n &= a + \left((x_n - a)^\alpha - \alpha \frac{g(x_n)}{(T_\alpha^a g)(x_n)}\right)^{\frac{1}{\alpha}}, \\ x_{n+1} &= a + \left((y_n - a)^\alpha - \alpha \left[\frac{g(x_n) + 2g(y_n)}{g(x_n)}\right] \frac{g(y_n)}{(T_\alpha^a g)(x_n)}\right)^{\frac{1}{\alpha}}. \end{aligned} \tag{3.3}$$

where, $(T_\alpha^a g)(x) = (x-a)^{1-\alpha} g'(x)$.

Problem 1. (Van Der Walls equation of state [12]) The Van der Waals equation, which serves as a refinement of the ideal gas law, is formulated as follows:

$$\left(p + \frac{n^2 a}{V^2}\right)(V - nb) = nRT. \tag{3.4}$$

Here, p, V, and T represent the pressure, volume, and temperature (in Kelvin), respectively, while n is the number of moles of the gas. The universal gas constant is denoted by R, with a value of approximately 0.0820578. The constants a and b, known as Van der Waals constants, are specific to the gas under consideration. The above equation is nonlinear with respect to V, and it can be reformulated into a cubic polynomial of the form:

$$g(V) = pV^3 - n(RT + bp)V^2 + n^2 aV - n^3 ab. \tag{3.5}$$

As an example, suppose we wish to determine the volume of 1.4 moles of benzene vapor under a pressure of 40 atm and a temperature of 500°C. Given that the Van der Waals constants for benzene are $a = 18$ and $b = 0.1154$, the equation becomes:

$$g_1(x) = 40x^3 - 95.26535116x^2 + 35.28x - 5.6998368. \tag{3.6}$$

This cubic equation has three roots, one of which is real: $x = 1.97078$, while the other two are complex: $x = 0.205425 \pm 0.173507i$. Since volume must be a real, positive quantity, only the real root is considered physically meaningful. For solving this equation numerically, the initial approximation $x_0 = 1.6$ was used.

Table 1. Results for $g_1(x)$, with initial estimate $x_0 = 1.6$.

	TeCO Method					**CKeCO Method**				
α	$\|g_1(x_{n+1})\|$	$\|x_{n+1} - x_n\|$	Iterations	NEF	ρ	$\|g_1(x_{n+1})\|$	$\|x_{n+1} - x_n\|$	Iterations	NFE	ρ
1	0	2.37×10^{-9}	16	3	3.08	1.35×10^{-11}	3.42×10^{-5}	126	3	2.76
0.9	7.11×10^{-14}	5.29×10^{-6}	16	3	2.81	1.36×10^{-12}	1.49×10^{-5}	11	3	2.79
0.8	4.83×10^{-13}	3.95×10^{-7}	15	3	2.87	5.80×10^{-14}	3.13×10^{-8}	25	3	2.95
0.7	7.11×10^{-14}	4.60×10^{-9}	11	3	2.93	7.11×10^{-14}	8.87×10^{-7}	14	3	2.86
0.6	6.25×10^{-10}	1.12×10^{-4}	61	3	3.07	5.79×10^{-14}	4.35×10^{-10}	18	3	2.94
0.5	2.56×10^{-13}	7.18×10^{-6}	28	3	2.81	4.53×10^{-9}	2.34×10^{-4}	46	3	2.68
0.4	3.82×10^{-12}	2.05×10^{-5}	23	3	2.69	2.56×10^{-13}	2.66×10^{-8}	10	3	2.92
0.3	8.21×10^{-11}	5.72×10^{-5}	337	3	2.65	7.11×10^{-14}	6.30×10^{-7}	8	3	2.87
0.2	2.55×10^{-13}	4.54×10^{-7}	18	3	2.87	2.55×10^{-13}	5.38×10^{-7}	6	3	3.22
0.1	-	-	>500	3	-	1.18×10^{-12}	7.45×10^{-7}	20	3	2.86
	CeCO Method					**NTJ Method**				
α	$\|g_1(x_{n+1})\|$	$\|x_{n+1} - x_n\|$	Iterations	NFE	ρ	$\|g_1(x_{n+1})\|$	$\|x_{n+1} - x_n\|$	Iterations	NFE	ρ
1	4.26×10^{-14}	4.83×10^{-6}	7	3	3.39	1.42×10^{-14}	3.42×10^{-5}	3	3	4.50
0.9	4.83×10^{-13}	2.06×10^{-5}	7	3	3.29	7.11×10^{-14}	3.51×10^{-5}	3	3	4.49
0.8	4.83×10^{-13}	7.88×10^{-5}	7	3	3.19	2.56×10^{-13}	3.59×10^{-5}	3	3	4.49
0.7	4.50×10^{-12}	2.71×10^{-4}	7	3	3.08	7.11×10^{-14}	3.67×10^{-5}	3	3	4.50
0.6	4.33×10^{-10}	8.39×10^{-4}	7	3	2.96	2.56×10^{-13}	3.76×10^{-5}	3	3	4.50
0.5	2.56×10^{-13}	2.11×10^{-10}	8	3	3.79	2.56×10^{-13}	3.84×10^{-5}	3	3	4.50
0.4	2.56×10^{-13}	8.06×10^{-9}	8	3	3.69	2.56×10^{-13}	3.93×10^{-5}	3	3	4.50
0.3	7.11×10^{-14}	2.41×10^{-7}	8	3	3.56	7.11×10^{-14}	4.03×10^{-5}	3	3	4.50
0.2	2.56×10^{-13}	4.56×10^{-6}	8	3	3.40	2.56×10^{-13}	4.12×10^{-5}	3	3	4.50
0.1	1.19×10^{-12}	5.96×10^{-5}	8	3	3.22	1.51×10^{-12}	4.21×10^{-5}	3	3	4.50

Problem 2. (Law of blood flow [17]) This problem is based on a well-known principle formulated by the French physiologist Jean Louis-Marie Poiseuille in 1840, which describes the flow of blood through cylindrical vessels such as veins and arteries. According to this law, the flow behavior

depends on the blood's viscosity η, the vessel's radius R, its length l, and the applied pressure P. The velocity profile is modeled by a function $g(x)$, where $x \in [0, R]$ represents the radial distance from the vessel's center.

The law yields the following nonlinear relationship:

$$g_2(x) = \frac{P}{\eta l}\left(R^2 - x^2\right). \tag{3.7}$$

For the purpose of numerical simulations, the parameters are taken as $P = 4000$, $\eta = 0.027$, $R = 0.008$, and $l = 2$. The function $g_2(x)$ is to be evaluated across the interval $x \in [0, R]$, representing the region within the blood vessel where flow velocity is analyzed.

TABLE 2. Results for $g_2(x)$, with initial estimate $x_0 = 0.9$.

	TeCO Method					**CKeCO Method**				
α	$\lvert g_2(x_{n+1})\rvert$	$\lvert x_{n+1} - x_n\rvert$	Iterations	NFE	ρ	$\lvert g_2(x_{n+1})\rvert$	$\lvert x_{n+1} - x_n\rvert$	Iterations	NFE	ρ
1	0	6.35×10^{-11}	8	3	2.92	0	6.35×10^{-11}	8	3	2.92
0.9	1.06×10^{-12}	7.04×10^{-11}	8	3	2.92	1.06×10^{-12}	7.21×10^{-11}	8	3	2.92
0.8	1.06×10^{-12}	7.79×10^{-11}	8	3	2.92	3.16×10^{-12}	8.17×10^{-12}	8	3	2.92
0.7	1.04×10^{-12}	8.62×10^{-11}	8	3	2.91	1.06×10^{-12}	9.24×10^{-11}	8	3	2.91
0.6	1.06×10^{-12}	9.53×10^{-11}	8	3	2.91	1.06×10^{-12}	1.04×10^{-10}	8	3	2.91
0.5	1.04×10^{-12}	1.05×10^{-10}	8	3	2.91	3.16×10^{-12}	1.18×10^{-10}	8	3	2.91
0.4	3.14×10^{-12}	1.16×10^{-10}	8	3	2.91	3.16×10^{-12}	1.33×10^{-10}	8	3	2.91
0.3	1.06×10^{-12}	1.28×10^{-10}	8	3	2.91	1.06×10^{-12}	1.50×10^{-10}	8	3	2.91
0.2	3.16×10^{-12}	1.41×10^{-10}	8	3	2.91	3.17×10^{-12}	1.69×10^{-10}	8	3	2.91
0.1	3.15×10^{-12}	1.56×10^{-10}	8	3	2.91	3.15×10^{-12}	1.90×10^{-10}	8	3	2.90
	CeCO Method					**NTJ Method**				
α	$\lvert g_2(x_{n+1})\rvert$	$\lvert x_{n+1} - x_n\rvert$	Iterations	NFE	ρ	$\lvert g_2(x_{n+1})\rvert$	$\lvert x_{n+1} - x_n\rvert$	Iterations	NFE	ρ
1	9.51×10^{-10}	2.86×10^{-5}	6	3	2.56	0	1.99×10^{-10}	6	3	3.63
0.9	1.08×10^{-9}	2.95×10^{-5}	6	3	2.55	1.06×10^{-12}	2.04×10^{-10}	6	3	3.63
0.8	1.22×10^{-9}	3.04×10^{-5}	6	3	2.55	1.04×10^{-12}	2.09×10^{-10}	6	3	3.63
0.7	1.37×10^{-9}	3.13×10^{-5}	6	3	2.54	1.06×10^{-12}	2.15×10^{-10}	6	3	3.62
0.6	1.54×10^{-9}	3.23×10^{-5}	6	3	2.54	1.06×10^{-12}	2.21×10^{-10}	6	3	3.62
0.5	1.74×10^{-9}	3.32×10^{-5}	6	3	2.53	1.04×10^{-12}	2.26×10^{-10}	6	3	3.62
0.4	1.95×10^{-9}	3.42×10^{-5}	6	3	2.53	3.15×10^{-12}	2.32×10^{-10}	6	3	3.62
0.3	2.20×10^{-9}	3.52×10^{-5}	6	3	2.52	3.15×10^{-12}	2.38×10^{-10}	6	3	3.62
0.2	2.47×10^{-9}	3.63×10^{-5}	6	3	2.52	5.25×10^{-12}	2.44×10^{-10}	6	3	3.62
0.1	2.77×10^{-9}	3.74×10^{-5}	6	3	2.51	3.15×10^{-12}	2.50×10^{-10}	6	3	3.62

Problem 3. (Chemical Engineering [14])

In chemical engineering, the acidity of a solution composed of magnesium hydroxide (MgOH) dissolved in hydrochloric acid (HCl) can be evaluated based on the concentration of hydronium ions, denoted as $[H_3O^+]$. The relationship governing this concentration is given by the equation:

$$\frac{3.64 \times 10^{-11}}{[H_3O^+]} = [H_3O^+] + 3.6 \times 10^{-4}.$$

To simplify the expression, let $x = 10^4[H_3O^+]$, resulting in the following nonlinear polynomial model:

$$g_3(x) = x^3 - 3.6x^2 - 36.4.$$

The solutions to this equation include two complex roots, $x = -3 + 2.3i$ and $x = -3 - 2.3i$, as well as one real root, $x = 2.4$, which is physically relevant in the context of acidity calculations. From the results corresponding to the NTJ method, it can be observed that when $\alpha = 1$, which represents the classical (non-fractional) case, the method requires 9 iterations to reach the desired tolerance. However, for fractional values of α, the NTJ method converges in fewer iterations compared to the classical case. This clearly indicates that the fractional version of the NTJ method improves convergence efficiency over its classical counterpart.

TABLE 3. Results for $g_3(x)$, with initial estimate $x_0 = 1.6$.

	TeCO Method					**CKeCO Method**				
α	$\lvert g_3(x_{n+1})\rvert$	$\lvert x_{n+1} - x_n\rvert$	Iterations	NFE	ρ	$\lvert g_3(x_{n+1})\rvert$	$\lvert x_{n+1} - x_n\rvert$	Iterations	NFE	ρ
1	7.11×10^{-15}	4.71×10^{-7}	24	3	3.16	4.63×10^{-10}	4.02×10^{-4}	96	3	3.88
0.9	6.11×10^{-14}	8.68×10^{-8}	14	3	2.93	5.90×10^{-14}	1.39×10^{-8}	15	3	3.01
0.8	1.19×10^{-13}	3.94×10^{-10}	13	3	3.02	4.67×10^{-11}	1.97×10^{-4}	13	3	2.74
0.7	9.04×10^{-9}	1.17×10^{-3}	12	3	2.70	8.22×10^{-12}	1.11×10^{-4}	13	3	2.78
0.6	1.17×10^{-13}	2.59×10^{-9}	19	3	2.96	1.19×10^{-13}	6.75×10^{-7}	27	3	2.91
0.5	5.71×10^{-10}	4.25×10^{-4}	12	3	2.74	4.97×10^{-14}	1.58×10^{-5}	13	3	2.84
0.4	6.25×10^{-14}	1.89×10^{-9}	13	3	2.99	6.33×10^{-12}	1.02×10^{-4}	13	3	2.78
0.3	7.48×10^{-11}	2.39×10^{-4}	11	3	2.72	1.76×10^{-13}	1.97×10^{-6}	12	3	2.96
0.2	1.95×10^{-11}	1.53×10^{-4}	26	3	2.75	1.49×10^{-13}	8.13×10^{-9}	151	3	2.96
0.1	-	-	>500	3	-	-	-	>500	3	-
	CeCO Method					**NTJ Method**				
α	$\lvert g_3(x_{n+1})\rvert$	$\lvert x_{n+1} - x_n\rvert$	Iterations	NFE	ρ	$\lvert g_3(x_{n+1})\rvert$	$\lvert x_{n+1} - x_n\rvert$	Iterations	NFE	ρ
1	7.11×10^{-15}	3.79×10^{-9}	35	3	3.76	7.11×10^{-15}	1.97×10^{-9}	9	3	3.81
0.9	5.90×10^{-14}	1.56×10^{-4}	18	3	4.55	1.21×10^{-12}	4.53×10^{-10}	8	3	4.14
0.8	1.19×10^{-13}	4.69×10^{-8}	18	3	3.78	6.83×10^{-10}	5.62×10^{-3}	8	3	3.01
0.7	1.14×10^{-13}	1.32×10^{-8}	19	3	3.76	7.11×10^{-15}	4.27×10^{-11}	6	3	3.87
0.6	1.17×10^{-13}	9.02×10^{-12}	63	3	4.11	7.82×10^{-14}	5.23×10^{-10}	5	3	4.14
0.5	4.97×10^{-14}	1.07×10^{-8}	22	3	3.74	7.82×10^{-14}	5.92×10^{-13}	5	3	3.92
0.4	679×10^{-14}	2.32×10^{-8}	25	3	3.83	7.11×10^{-15}	5.38×10^{-5}	5	3	3.43
0.3	1.19×10^{-13}	1.38×10^{-9}	29	3	3.85	2.63×10^{-13}	6.54×10^{-8}	6	3	3.73
0.2	1.19×10^{-13}	2.92×10^{-6}	124	3	3.68	2.63×10^{-13}	4.03×10^{-9}	7	3	3.80
0.1	1.21×10^{-13}	7.90×10^{-13}	50	3	3.90	5.04×10^{-10}	5.10×10^{-3}	8	3	3.03

Problem 4. (Law of population growth [17]) In population dynamics, a widely used mathematical representation is the first-order linear ordinary differential equation given by:

$$P'(t) = \kappa P(t) + v, \tag{3.8}$$

where $P(t)$ represents the population at time t, κ is the constant birth rate, and v is the constant immigration rate. Solving this differential equation yields the general solution:

$$P(t) = P_0 \exp(\kappa t) + \frac{v}{\kappa}(\exp(\kappa t) - 1), \tag{3.9}$$

in which P_0 denotes the initial population size. By applying a specific initial condition and parameter values provided in [8], the value of the birth rate κ can be determined through the following nonlinear equation:

$$g_4(x) = 1564 - 100 \exp(x) - \frac{435}{x}(\exp(x) - 1) = 0, \tag{3.10}$$

where $x = \kappa$ is the birth rate to be computed. From Table 4, it is noteworthy that the NTJ method achieves the prescribed tolerance within only two iterations for $\alpha = 0.6$ and $\alpha = 0.2$, demonstrating its accelerated convergence behavior. In contrast, for $\alpha = 1$ (the classical case), the method requires three iterations to reach the same level of accuracy. This clearly indicates that the fractional-order formulation of the NTJ method enhances convergence efficiency compared to its classical counterpart.

TABLE 4. Results for $g_4(x)$, with initial estimate $x_0 = 2$.

	TeCO Method					**CKeCO Method**				
α	$\|g_4(x_{n+1})\|$	$\|x_{n+1} - x_n\|$	Iterations	NFE	ρ	$\|g_4(x_{n+1})\|$	$\|x_{n+1} - x_n\|$	Iterations	NFE	ρ
1	2.85×10^{-15}	1.89×10^{-6}	3	3	2.94	1.61×10^{-14}	3.18×10^{-6}	3	3	2.93
0.9	4.66×10^{-12}	2.02×10^{-6}	3	3	2.94	4.66×10^{-12}	3.41×10^{-6}	3	3	2.93
0.8	5.25×10^{-13}	2.15×10^{-6}	3	3	2.94	5.25×10^{-13}	3.64×10^{-6}	3	3	2.93
0.7	1.54×10^{-12}	2.29×10^{-6}	3	3	2.94	1.54×10^{-12}	3.89×10^{-6}	3	3	2.93
0.6	1.54×10^{-12}	2.44×10^{-6}	3	3	2.94	1.54×10^{-12}	4.16×10^{-6}	3	3	2.93
0.5	5.25×10^{-13}	2.60×10^{-6}	3	3	2.94	5.25×10^{-13}	4.43×10^{-6}	3	3	2.93
0.4	1.54×10^{-12}	2.77×10^{-6}	3	3	2.93	1.54×10^{-12}	4.73×10^{-6}	3	3	2.92
0.3	1.54×10^{-12}	2.94×10^{-6}	3	3	2.93	1.54×10^{-12}	5.04×10^{-6}	3	3	2.92
0.2	2.59×10^{-12}	3.12×10^{-6}	3	3	2.93	2.59×10^{-12}	5.36×10^{-6}	3	3	2.92
0.1	5.68×10^{-12}	3.32×10^{-6}	3	3	2.93	5.68×10^{-12}	5.70×10^{-6}	3	3	2.92
	CeCO Method					**NTJ Method**				
α	$\|g_4(x_{n+1})\|$	$\|x_{n+1} - x_n\|$	Iterations	NFE	ρ	$\|g_4(x_{n+1})\|$	$\|x_{n+1} - x_n\|$	Iterations	NFE	ρ
1	9.43×10^{-37}	2.22×10^{-10}	3	3	3.88	7.02×10^{-57}	2.48×10^{-15}	3	3	3.99
0.9	4.66×10^{-12}	2.49×10^{-10}	3	3	3.88	4.66×10^{-12}	3.55×10^{-15}	3	3	3.94
0.8	2.59×10^{-12}	2.80×10^{-10}	3	3	3.88	1.54×10^{-12}	3.55×10^{-15}	3	3	3.95
0.7	5.25×10^{-13}	3.13×10^{-10}	3	3	3.88	5.25×10^{-13}	3.55×10^{-15}	3	3	3.96
0.6	1.54×10^{-12}	3.50×10^{-10}	3	3	3.88	2.59×10^{-12}	5.95×10^{-4}	2	3	-
0.5	5.25×10^{-13}	3.91×10^{-10}	3	3	3.87	5.25×10^{-13}	1.78×10^{-15}	3	3	4.08
0.4	1.54×10^{-12}	4.37×10^{-10}	3	3	3.87	1.54×10^{-12}	5.33×10^{-15}	3	3	3.91
0.3	1.54×10^{-12}	4.87×10^{-10}	3	3	3.87	1.54×10^{-12}	8.88×10^{-15}	3	3	3.84
0.2	2.59×10^{-12}	5.42×10^{-10}	3	3	3.87	2.59×10^{-12}	6.15×10^{-4}	2	3	-
0.1	5.68×10^{-12}	6.02×10^{-10}	3	3	3.87	5.68×10^{-12}	2.13×10^{-14}	3	3	3.92

Problem 5. (Planck's radiation law [11]) Let us consider Planck's law of blackbody radiation

as presented in [11]:

$$\gamma = \frac{8\pi ch}{\lambda^5 \left(e^{\frac{ch}{\lambda kT}} - 1\right)},$$

where γ represents the spectral energy density, λ is the wavelength, T is the temperature in Kelvin, k is Boltzmann's constant, h is Planck's constant, and c is the speed of light in vacuum. To identify the wavelength λ at which the energy density attains its maximum, we take the derivative of γ with respect to λ:

$$\frac{d\gamma}{d\lambda} = \frac{8\pi ch}{\lambda^5 \left(e^{\frac{ch}{\lambda kT}} - 1\right)} \left(-5 + \frac{\left(\frac{ch}{\lambda kT}\right) e^{\frac{ch}{\lambda kT}}}{e^{\frac{ch}{\lambda kT}} - 1}\right).$$

As λ approaches 0 or ∞, the expression inside the parentheses tends toward zero, and γ reaches a minimum at both extremes. The desired maximum occurs when the expression within the parentheses is equal to zero, leading to:

$$\frac{\left(\frac{ch}{\lambda_{\max} kT}\right) e^{\frac{ch}{\lambda_{\max} kT}}}{e^{\frac{ch}{\lambda_{\max} kT}} - 1} = 5,$$

where $\lambda_{\max}$ represents the wavelength corresponding to the peak energy density. By letting $x = \frac{ch}{\lambda_{\max} kT}$, the equation simplifies to:

$$1 - \frac{x}{5} = e^{-x}.$$

Now we can define the following non-linear equation,

$$g_5(x) = e^{-x} - 1 + \frac{x}{5}.$$

4. Numerical stability of the Developed method

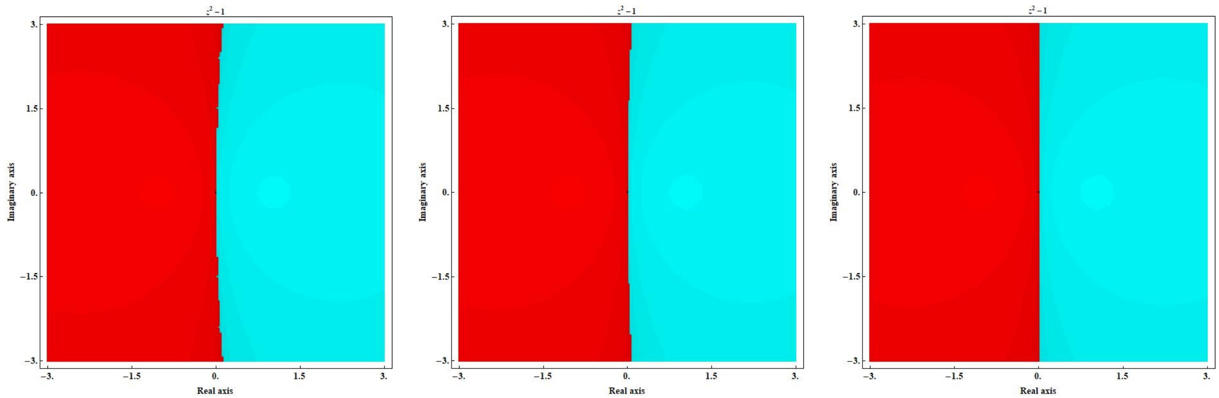
In their investigation of the fractional Newton method's stability, Akgül et al. [1] employed a graphical approach derived from Cayley's quadratic test. This technique utilizes polynomiographs to illustrate how different regions are influenced by the roots of polynomial functions, depending on the fractional order α. These visualizations offer a deeper understanding of how the method behaves in terms of stability across varying values of α.

To show the basins of attraction, we selected a rectangular region D within the complex plane $\mathbb{C}$. We chose the area $[-3, 3] \times [-3, 3]$, which contains all the roots of the equation $Q(z) = 0$. A grid of 200×200 points was used to create a detailed picture. Each point on the grid represents a starting value. In the basin of attraction analysis, a point z_0 is considered to belong to the basin of a root z^* of the polynomial if the iterative method converges to z^* within a tolerance of $|z_k - z^*| < 10^{-3}$ and within a maximum of 25 iterations. Such points are visually represented using distinct colors corresponding to the associated roots. Within each basin, the shade of the color indicates the speed of convergence—brighter shades represent fewer iterations, while darker shades signify a higher number of iterations required. Points that either do not converge within the iteration limit or diverge to infinity are marked in black, indicating no successful convergence.

TABLE 5. Results for $g_5(x)$, with initial estimate $x_0 = -2.5$.

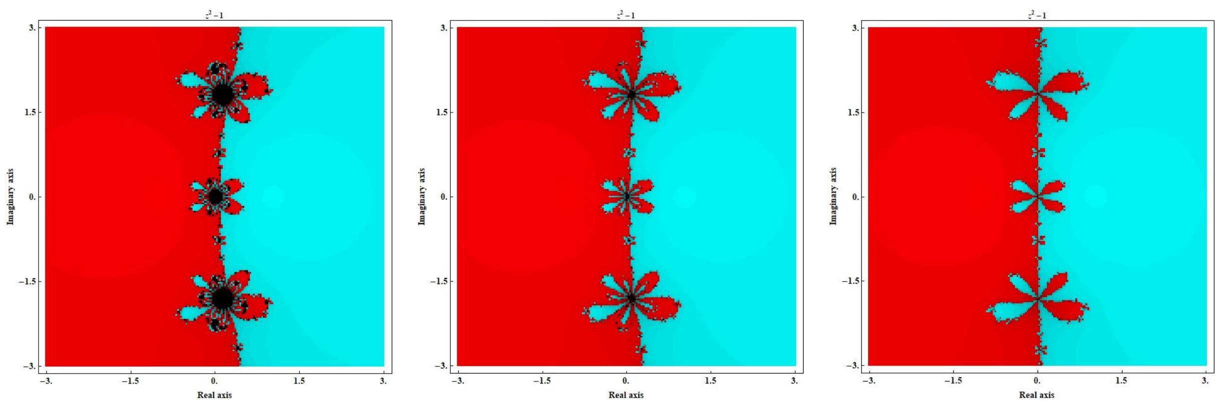
	TeCO Method					CKeCO Method				
α	$\lvert g_5(x_{n+1})\rvert$	$\lvert x_{n+1}-x_n\rvert$	Iterations	NFE	ρ	$\lvert g_5(x_{n+1})\rvert$	$\lvert x_{n+1}-x_n\rvert$	Iterations	NFE	ρ
1	3.32×10^{-15}	1.74×10^{-5}	5	3	2.88	1.25×10^{-13}	5.61×10^{-5}	5	3	2.86
0.9	0	1.44×10^{-5}	5	3	2.88	6.82×10^{-14}	4.63×10^{-5}	5	3	2.87
0.8	1.42×10^{-15}	1.78×10^{-15}	6	3	2.95	3.69×10^{-14}	3.80×10^{-5}	5	3	2.87
0.7	1.42×10^{-15}	9.65×10^{-6}	5	3	2.89	1.99×10^{-14}	3.10×10^{-5}	5	3	2.87
0.6	1.42×10^{-15}	7.85×10^{-6}	5	3	2.89	9.95×10^{-15}	2.52×10^{-5}	5	3	2.87
0.5	1.42×10^{-15}	6.35×10^{-6}	5	3	2.89	5.68×10^{-15}	2.03×10^{-5}	5	3	2.80
0.4	0	5.12×10^{-6}	5	3	2.90	2.84×10^{-15}	1.63×10^{-5}	5	3	2.88
0.3	0	4.10×10^{-6}	5	3	2.90	0	1.30×10^{-5}	5	3	2.88
0.2	2.84×10^{-15}	3.27×10^{-6}	5	3	2.90	2.84×10^{-15}	1.03×10^{-5}	5	3	2.89
0.1	4.26×10^{-15}	2.59×10^{-6}	5	3	2.90	4.26×10^{-15}	8.14×10^{-6}	5	3	2.89
	CeCO Method					**NTJ Method**				
α	$\lvert g_5(x_{n+1})\rvert$	$\lvert x_{n+1}-x_n\rvert$	Iterations	NFE	ρ	$\lvert g_5(x_{n+1})\rvert$	$\lvert x_{n+1}-x_n\rvert$	Iterations	NFE	ρ
1	3.31×10^{-13}	7.74×10^{-4}	4	3	3.41	8.50×10^{-10}	9.85×10^{-3}	3	3	3.72
0.9	1.85×10^{-13}	6.84×10^{-4}	4	3	3.42	8.15×10^{-10}	0.00976	3	3	3.72
0.8	1.08×10^{-13}	6.03×10^{-4}	4	3	3.42	7.78×10^{-10}	9.07×10^{-3}	3	3	3.73
0.7	6.39×10^{-14}	5.29×10^{-4}	4	3	3.43	7.41×10^{-10}	9.57×10^{-3}	3	3	3.73
0.6	3.55×10^{-14}	4.63×10^{-4}	4	3	3.43	7.02×10^{-10}	9.46×10^{-3}	3	3	3.73
0.5	2.13×10^{-14}	4.05×10^{-4}	4	3	3.43	6.64×10^{-10}	9.35×10^{-3}	3	3	3.73
0.4	9.95×10^{-15}	3.52×10^{-4}	4	3	3.44	6.25×10^{-10}	9.22×10^{-3}	3	3	3.73
0.3	5.68×10^{-15}	3.05×10^{-4}	4	3	3.44	5.86×10^{-10}	9.09×10^{-3}	3	3	3.73
0.2	2.84×10^{-15}	2.64×10^{-4}	4	3	3.45	5.48×10^{-10}	8.96×10^{-3}	3	3	3.73
0.1	4.26×10^{-15}	2.27×10^{-4}	4	3	3.45	5.10×10^{-10}	8.82×10^{-3}	3	3	3.73

Example 1. As a first test case, we consider the polynomial function $Q_1(z) = z^2 - 1$. This is a simple quadratic equation with two real roots, namely $z = 1$ and $z = -1$. The basins of attraction depicted in Figures 1-4 illustrate the convergence behavior of the considered methods for $Q_1(z)$ across different values of α. It is clearly observed that the NTJ method (Figure 1) exhibits a highly regular and well-structured basin, characterized by a sharp and smooth boundary separating the regions of convergence. Unlike the other methods, NTJ shows almost no scattered or chaotic regions, indicating strong stability and minimal sensitivity to initial guesses. In contrast, CKeCo, CeCo, and TeCo display irregular patterns with noticeable fractal boundaries and black regions, reflecting divergence or slow convergence. Furthermore, the basin corresponding to NTJ remains nearly unchanged as α varies, highlighting its robustness. Overall, the graphical evidence strongly supports that the NTJ method provides superior convergence behavior with enhanced stability and reliability.



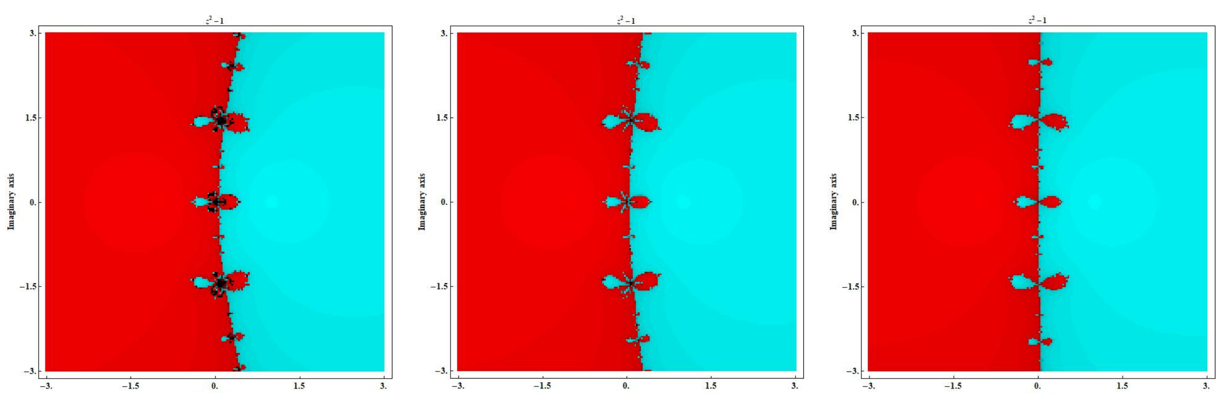
(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

FIGURE 1. Basins of attraction for $Q_1(z)$ by using NTJ



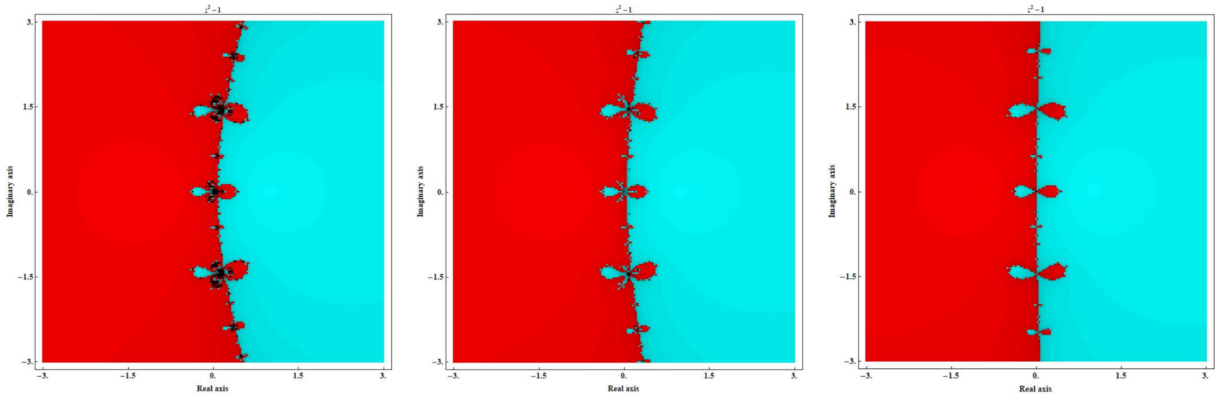
(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

FIGURE 2. Basins of attraction for $Q_1(z)$ by using CeCo



(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

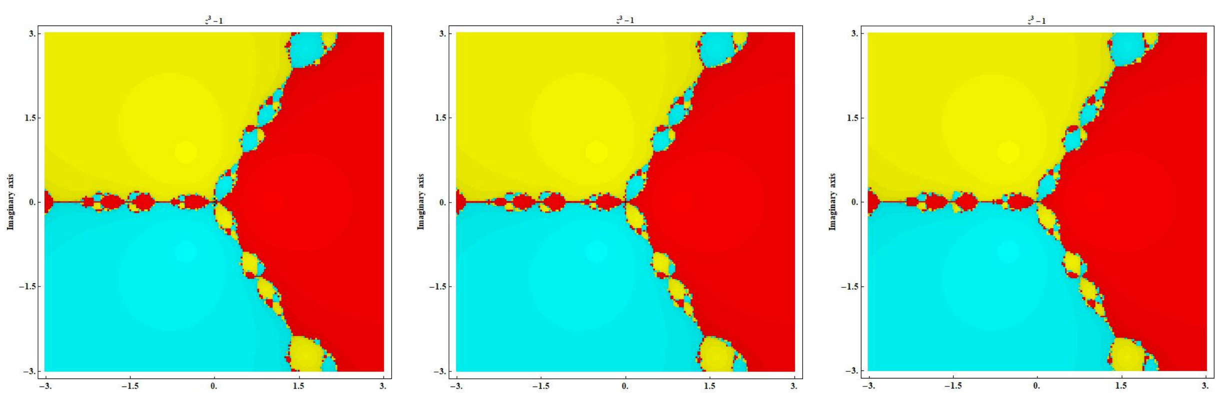
FIGURE 3. Basins of attraction for $Q_1(z)$ by using TeCo



(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

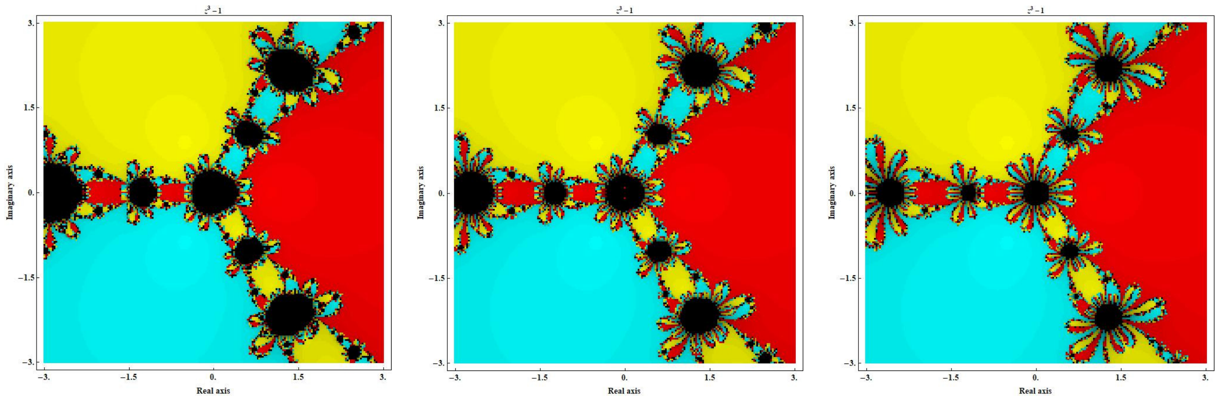
FIGURE 4. Basins of attraction for $Q_1(z)$ by using CKeCo

Example 2. Next, we consider the polynomial $Q_2(z) = z^3 - 1$, which has three complex roots: $z = 1$, $z = -\frac{1}{2} + \frac{\sqrt{3}}{2}i$, and $z = -\frac{1}{2} - \frac{\sqrt{3}}{2}i$. The basins of attraction for $Q_2(z)$ illustrated in Figures 5–8 further emphasize the superior performance of the NTJ method. As seen in Figure 5, the NTJ method produces smooth, well-defined, and symmetric basins with very few irregular or chaotic regions, indicating strong stability and reliable convergence. The boundaries between attraction regions are clear and exhibit minimal fractal behavior, even as α varies. In contrast, the basins corresponding to CeCo, TeCo, and CKeCo (Figures 6–8) display highly intricate fractal structures with significant scattering and numerous black regions, suggesting sensitivity to initial guesses and a higher likelihood of divergence. Additionally, the NTJ basins remain nearly unchanged across different values of α, demonstrating robustness and consistency. These observations confirm that the NTJ method ensures faster, more stable, and dependable convergence compared to the competing methods.



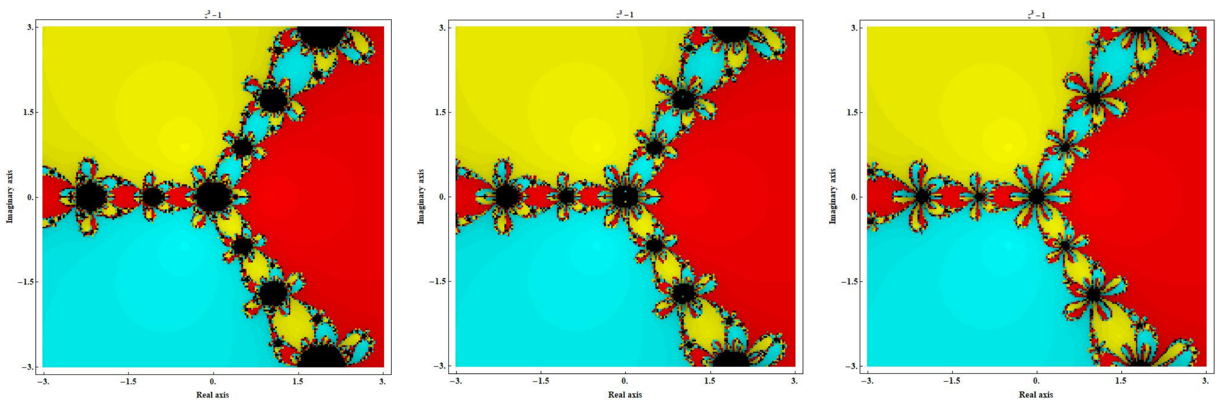
(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

FIGURE 5. Basins of attraction for $Q_2(z)$ by using NTJ



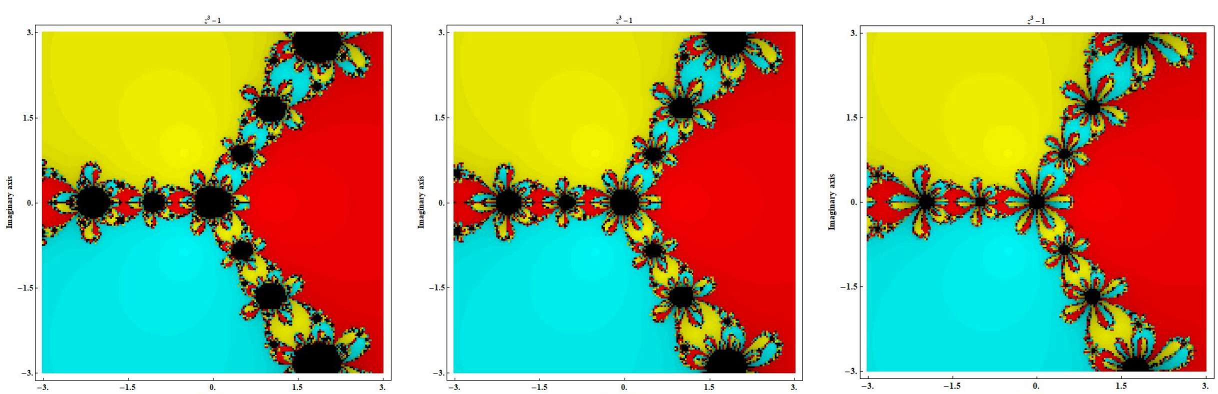
(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

FIGURE 6. Basins of attraction for $Q_2(z)$ by using CeCo



(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

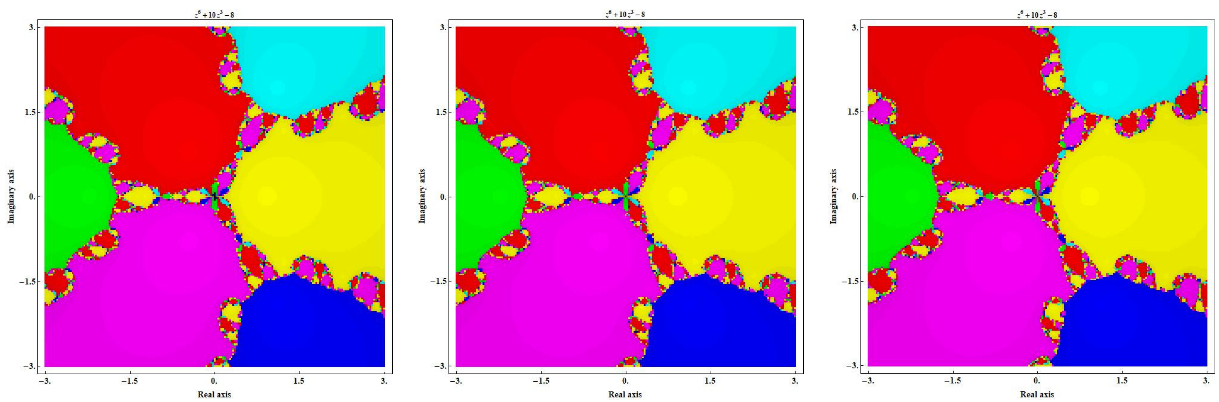
FIGURE 7. Basins of attraction for $Q_2(z)$ by using TeCo



(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

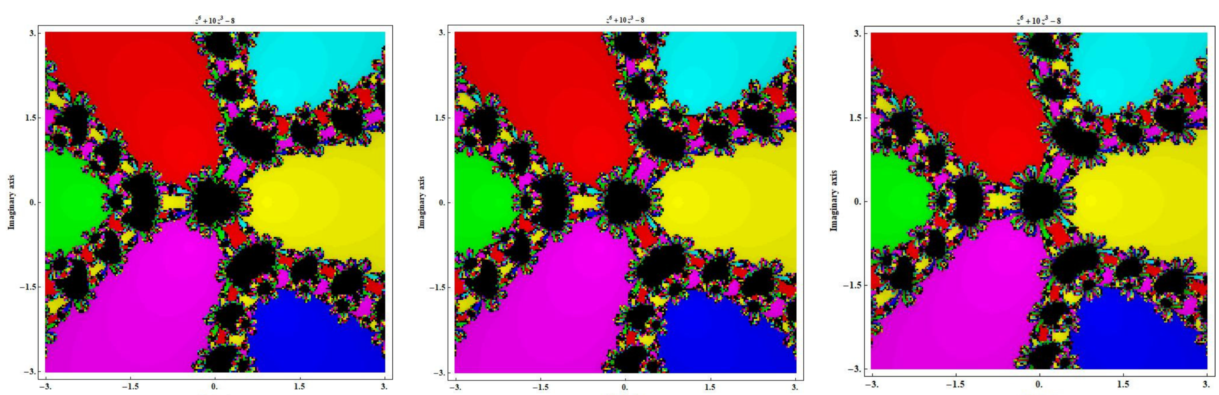
FIGURE 8. Basins of attraction for $Q_2(z)$ by using CKeCo

Example 3. As a third example, we analyze the polynomial $Q_3(z) = z^6 - 10z^3 - 8$, which has six complex roots. These include $z = 0.906359, z = -2.20663$, $z = 1.103 \pm 1.911i$, and $z = -0.453 \pm 0.785i$. From Figures 9–12, it is evident that the NTJ method provides a more structured and stable convergence pattern for $Q_3(z)$. The attraction regions in Figure 9 are wider and more uniformly distributed, with sharp and less chaotic boundaries. On the other hand, competing methods such as CeCo, TeCo, and CKeCo generate irregular and highly fractal basin structures, along with increased non-convergent (black) regions. The minimal variation in NTJ basins with respect to α further demonstrates its consistency and reduced sensitivity to parameter changes.



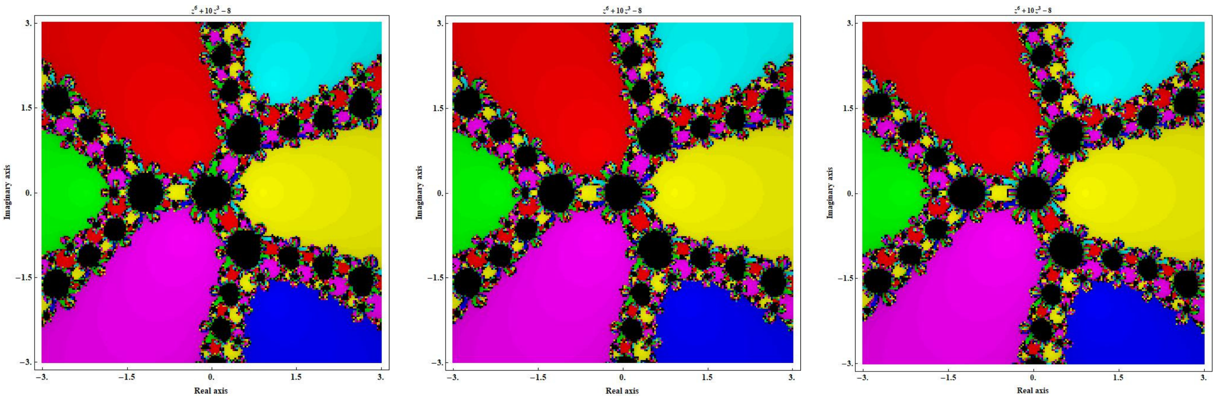
(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

FIGURE 9. Basins of attraction for $Q_3(z)$ by using NTJ



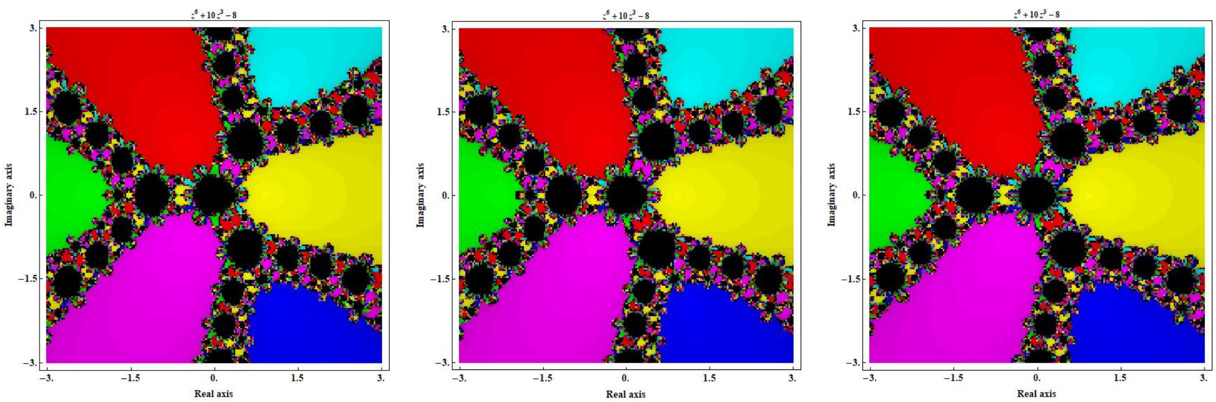
(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

FIGURE 10. Basins of attraction for $Q_3(z)$ by using CeCo



(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

Figure 11. Basins of attraction for $Q_3(z)$ by using TeCo



(A) $\alpha = 0.2$ (B) $\alpha = 0.5$ (C) $\alpha = 0.9$

Figure 12. Basins of attraction for $Q_3(z)$ by using CKeCo

The results presented in Tables 6–8 provide a comprehensive comparison of the considered methods in terms of black points, average number of iterations, and CPU time for the test functions $Q_1(z)$, $Q_2(z)$, and $Q_3(z)$. It is evident that the NTJ method significantly outperforms the other methods across all values of α. In particular, NTJ consistently produces the smallest number of black points, indicating a larger region of convergence and higher stability. Moreover, it achieves the lowest average number of iterations, demonstrating its faster convergence rate. The computational efficiency of NTJ is further supported by its reduced CPU time in most cases. In contrast, the other methods exhibit higher iteration counts, increased computational cost, and a larger number of divergent points. Overall, these results clearly confirm the robustness, efficiency, and superior performance of the NTJ method over the competing schemes.

Figure 13 compares the average number of iterations for NTJ and the existing methods (TeCo, CKeCo, and CeCo) over $\alpha \in [0.1, 0.9]$. It is observed that the NTJ method consistently requires the fewest iterations and remains nearly constant across all values of α. In contrast, the other methods show an increasing trend in iterations as α grows, indicating slower convergence and reduced stability. These results highlight the superior efficiency and robustness of the NTJ method.

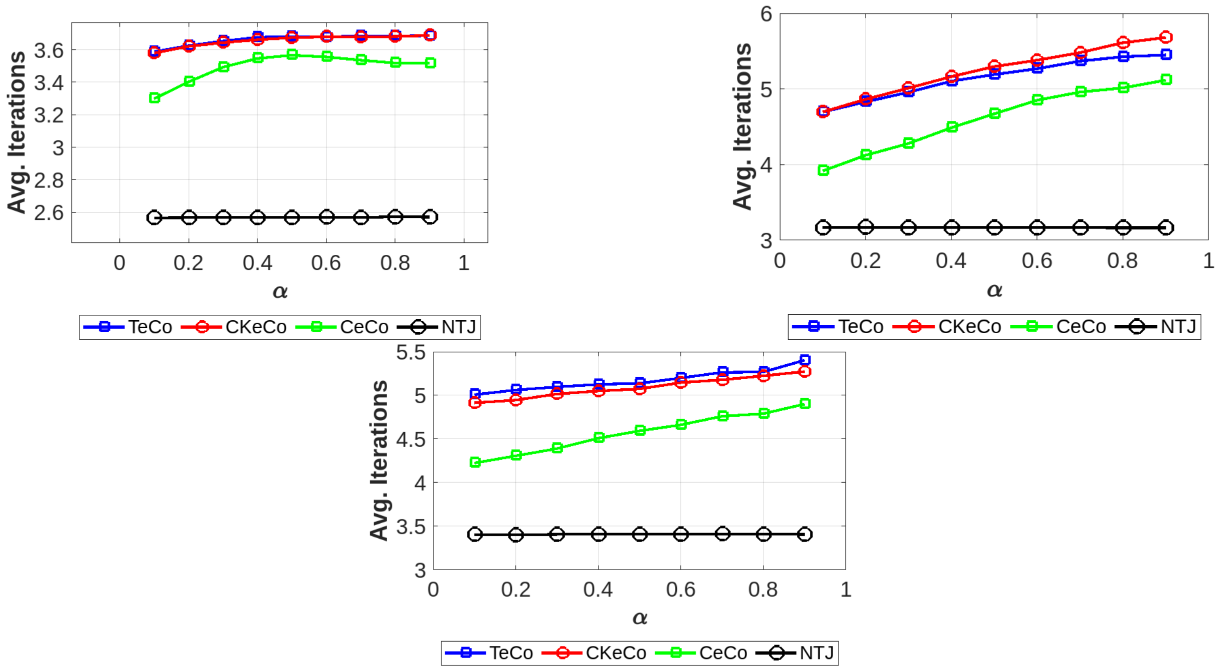


Figure 13. Comparison of the average number of iterations for different methods with varying values of the fractional parameter α.

Table 6. Comparison of methods based on number of black points, average iterations, and CPU time for $Q_1(z)$.

Function	$\boldsymbol{\alpha}$	**Method**	**Black Points**	**Average No. of Iterations**	**CPU Time**
$Q_1(z)$	0.2	TeCo	199	3.6246	16.719
		CKeCo	201	3.6205	17.781
		CeCo	751	3.4019	17.703
		NTJ	1	2.5672	12.375
	0.5	TeCo	17	3.6806	15.625
		CKeCo	17	3.6740	17.156
		CeCo	104	3.5661	17.438
		NTJ	1	2.5675	12.125
	0.9	TeCo	1	3.6862	15.391
		CKeCo	1	3.6881	17.328
		CeCo	3	3.5163	17.531
		NTJ	1	2.5706	13.047

TABLE 7. Comparison of methods based on number of black points, average iterations, and CPU time for $Q_2(z)$.

Function	α	**Method**	**Black Points**	**Average No. of Iterations**	**CPU Time**
$Q_2(z)$	0.2	TeCo	2903	4.8295	28.39
		CKeCo	3793	4.8635	35.188
		CeCo	4602	4.1237	35.344
		NTJ	4	3.1714	15.047
	0.5	TeCo	1992	5.1907	27.781
		CKeCo	2666	5.2964	33.218
		CeCo	3214	4.6715	33.156
		NTJ	1	3.1689	15.86
	0.9	TeCo	1086	5.4495	26.969
		CKeCo	1537	5.6802	31.781
		CeCo	2034	5.1166	32.062
		NTJ	1	3.1653	16.093

TABLE 8. Comparison of methods based on number of black points, average iterations, and CPU time for $Q_3(z)$.

Function	α	**Method**	**Black Points**	**Average No. of Iterations**	**CPU Time**
$Q_3(z)$	0.2	TeCo	6450	5.0615	55.172
		CKeCo	8117	4.9457	66.531
		CeCo	8274	4.3074	68.953
		NTJ	16	3.4019	23.781
	0.5	TeCo	6318	5.1379	2.859
		CKeCo	7873	5.0738	63.453
		CeCo	7768	4.5936	64.078
		NTJ	4	3.4092	24.000
	0.9	TeCo	5818	5.4028	52.016
		CKeCo	7534	5.2733	64.750
		CeCo	7037	4.9006	63.110
		NTJ	1	3.4071	24.328

5. CONCLUSION

In this study, we have developed a novel two-step fractional iterative method by incorporating the conformable derivative into an existing classical two-step scheme. The convergence behavior of the proposed method has been rigorously analyzed, and it has been established that the method achieves fourth-order convergence. Through various real-world examples and comparative numerical experiments, the method has demonstrated superior performance, particularly in terms of reducing the number of iterations required. Furthermore, the basin of attraction analysis reinforced the method's robustness and efficiency in solving nonlinear equations. The proposed approach represents a significant improvement over traditional techniques and offers a promising alternative for tackling nonlinear problems with greater accuracy and stability.

Acknowledgement: The authors are very much thankful to the Department of Science and Technology, New Delhi, India for approving the proposal under the scheme FIST program (Ref. No. SR/FST/MS/2022 dated 19.12.2022).

Competing interests: The authors declare that there is no conflict of interest regarding the publication of this paper.

References

[1] A. Akgül, A. Cordero, J.R. Torregrosa, A fractional Newton method with 2αth-order of convergence and its stability, Appl. Math. Lett. 98 (2019), 344–351.
[2] A. Cordero, J.R. Torregrosa, Variants of Newton's method using fifth-order quadrature formulas, Appl. Math. Comput. 190 (2007), 686–698.
[3] A.M. Mathai, H.J. Haubold, Fractional and Multivariable Calculus: Model Building and Optimization Problems, Springer, 2017.
[4] A. Torres-Hernandez, F. Brambila-Paz, U. Iturrarán-Viveros, R. Caballero-Cruz, Fractional Newton–Raphson method accelerated with Aitken's method, Axioms 10 (2021), 47.
[5] G. Candelario, A. Cordero, J.R. Torregrosa, M.P. Vassileva, An optimal and low computational cost fractional Newton-type method for solving nonlinear equations, Appl. Math. Lett. 124 (2022), 107650.
[6] G. Candelario, A. Cordero, J.R. Torregrosa, M.P. Vassileva, Generalized conformable fractional Newton-type method for solving nonlinear systems, Numer. Algorithms 93 (2023), 1171–1208.
[7] G. Candelario, A. Cordero, J.R. Torregrosa, M.P. Vassileva, Solving nonlinear transcendental equations by iterative methods with conformable derivatives: A general approach, Mathematics 11 (2023), 2568.
[8] J.R. Sharma, S. Kumar, H. Singh, A new class of derivative-free root solvers with increasing optimal convergence order and their complex dynamics, SeMA J. 80 (2023), 333–352.
[9] K. Gdawiec, W. Kotarski, A. Lisowska, Newton's method with fractional derivatives and various iteration processes via visual analysis, Numer. Algorithms 86 (2021), 953–1010.
[10] K.S. Miller, B. Ross, An Introduction to the Fractional Calculus and Fractional Differential Equations, John Wiley & Sons, 1993.
[11] M.D. Junjua, S. Abdullah, M. Kansal, S. Ahmad, On Traub–Steffensen-type iteration schemes with and without memory: Fractal analysis using basins of attraction, Fractal Fract. 8 (2024), 698.
[12] O. Said Solaiman, I. Hashim, Efficacy of optimal methods for nonlinear equations with chemical engineering applications, Math. Probl. Eng. 2019 (2019), 1728965.
[13] R. Khalil, M. Al Horani, A. Yousef, M. Sababheh, A new definition of fractional derivative, J. Comput. Appl. Math. 264 (2014), 65–70.
[14] S.C. Chapra, Applied Numerical Methods with MATLAB for Engineers and Scientists, McGraw-Hill, 2018.
[15] S. Li, Fourth-order iterative method without calculating the higher derivatives for nonlinear equation, J. Algorithms Comput. Technol. 13 (2019), 1–8.
[16] S.K. Nayak, P.K. Parida, The dynamical analysis of a low computational cost family of higher-order fractional iterative method, Int. J. Comput. Math. 100 (2023), 1395–1417.

[17] S. Qureshi, A. Soomro, A.A. Shaikh, E. Hincal, N. Gokbulut, A novel multistep iterative technique for models in medical sciences with complex dynamics, Comput. Math. Methods Med. 2022 (2022), 7656451.

[18] Ş. Toprakseven, Numerical solutions of conformable fractional differential equations by Taylor and finite difference methods, Süleyman Demirel Univ. Fen Bilim. Enst. Derg. 23 (2019), 850–863.

[19] T. Abdeljawad, On conformable fractional calculus, J. Comput. Appl. Math. 279 (2015), 57–66.