



Research article

On the stability analysis of numerical schemes for solving non-linear polynomials arises in engineering problems

Mudassir Shams^{1,2}, Nasreen Kausar³, Serkan Araci⁴ and Liang Kong^{5,*}

¹ Faculty of Engineering, Free University of Bozen-Bolzano (BZ), 39100, Italy

² Department of Mathematics and Statistics, Riphah International University I-14, Islamabad 44000, Pakistan

³ Department of Mathematics, Faculty of Arts and Science, Yildiz Technical University, Esenler 34220, Istanbul, Türkiye

⁴ Department of Basic Sciences, Faculty of Engineering, Hasan Kalyoncu University, TR-27010 Gaziantep, Türkiye

⁵ Department of Mathematical Sciences and Philosophy, University of Illinois Springfield, USA

* **Correspondence:** Email: lkong9@uis.edu.

Abstract: This study shows the link between computer science and applied mathematics. It conducts a dynamics investigation of new root solvers using computer tools and develops a new family of single-step simple root-finding methods. The convergence order of the proposed family of iterative methods is two, according to the convergence analysis carried out using symbolic computation in the computer algebra system CAS-Maple 18. Without further evaluations of a given nonlinear function and its derivatives, a very rapid convergence rate is achieved, demonstrating the remarkable computing efficiency of the novel technique. To determine the simple roots of nonlinear equations, this paper discusses the dynamic analysis of one-parameter families using symbolic computation, computer animation, and multi-precision arithmetic. To choose the best parametric value used in iterative schemes, it implements the parametric and dynamical plane technique using CAS-MATLAB® R2011b. The dynamic evaluation of the methods is also presented utilizing basins of attraction to analyze their convergence behavior. Aside from visualizing iterative processes, this method illustrates not only iterative processes but also gives useful information regarding the convergence of the numerical scheme based on initial guessed values. Some nonlinear problems that arise in science and engineering are used to demonstrate the performance and efficiency of the newly developed method compared to the existing method in the literature.

Keywords: complex dynamics; parametric plane; stability region; cpu time; engineering application

Mathematics Subject Classification: 65H04, 65H05, 65Y05, 65M12

1. Introduction

One of the most critical areas of applied mathematics are iterative techniques to solve nonlinear equations. Their importance is due to their wide applications in different fields, such as engineering, computer science, physics, biology, chemistry, banking, business, digital signal processing, control theory, insurance, social science, and many others. For instance, [1–4] evidence some of these applications. Computer visual representation, which converts numerical data into 2D or 3D images, unquestionably offers a modern and unconventional solution to some mathematical topics. It gives a fresh perspective and provides a deep knowledge of the characteristics and behavior of iterative processes for finding polynomial roots and allocating them (e.g., [5–11]). The development of digital computer hardware and software has substantially impacted many scientific disciplines and is frequently utilized in almost all scientific domains, including mathematics. The usage of advanced computational tools, such as symbolic computation and computer graphics, is necessary to tackle many complex problems in applied mathematics and other scientific disciplines. Thanks to the rapid development of computer graphics over the past 20 years, scientists from various fields can now use cutting-edge computer techniques to study and visualize phenomena that are not easily observable. Architecture and engineering, biomedical applications (X-ray computed tomography (CT), magnetic resonance imaging (MRI), mathematics, physics, business and management graphics, education and learning, user interface design, image syntheses, mapping and cartography, video and multimedia technology (animation, move, etc.), and business are just a few examples where high-quality computer visualization provides new and remarkable insights. Many problems in the areas above remain unsolvable by humans. Computer graphics allow a quick and practical link between machines and people. The technology and mathematics that computer graphics provide to a wide range of applications in theory and practice, and they indeed have great potential in the future (see e.g., [12–16]). The initial guess value affects the classical Newton-like method's performance, particularly for large-scale nonlinear problems. For this reason, many iterations of Newton's technique have been devised. For example, Knoll et al. [17] suggested the arc-length approaches, and Lemieux et al. [18] offered the Jacobian-Free Newton-Krylov method. Wu et al. [19] provided novel contributions to Newton's research. Qureshi et al. [20] presents the trapezoidal second-order Newton's, He et al. [21] presented the variational Newton-type scheme, Saheya et al. [22] employed the rational approximation function to suggest a Newton-like method, Abbasbandy [23] proposed an improved version of Newton's scheme by using a modified Adomian decomposition technique. Motivated by the above mentioned work, this study develops a novel family of Newton-type iterative algorithms using a single parameter for locating the simple root of nonlinear equations. The parameter value influences the newly developed family. The best parameter value for that family of methods should be chosen based on the method that has a larger region of convergence at starting points where the classical methods diverge. To achieve this aim, we deal with the corresponding complex dynamical system, determining fixed and critical points, analyzing asymptotic convergence, and identifying values that decompose the corresponding rational function. The main contribution of this work is reflected as follows:

- Construction of a family of numerical schemes for finding roots of scalar nonlinear equations.
- Using CAS-Maple to verify the convergence order of the proposed method.
- Finding fixed and critical points utilizing complex dynamical analysis.

- Dynamical and parametric planes are generated using critical points.
- Computational tools are used to analyze the proposed technique's convergence rate, efficiency, stability, and applicability.
- Analyzing how the suggested technique can be applied in various research disciplines.

Some engineering applications are considered to show improved efficiency and stability compared to other classical techniques discussed in the literature. This research aims to construct a novel family of iterative methods using the computer algebra system CAS-Maple, Mathematica, MATLAB, Scilab, GNU Octave, Python, and many others (see e.g., [24, 25]). It uses parameter and dynamical planes to apply these techniques for picking the best parameter value for the iterative scheme. This paper is organized as follows: In section 2, the proposed method is introduced. In section 3, the stability of the proposed method using complex dynamical analysis is discussed. In section 4, the efficiency of this method is illustrated by applications. The last section is the conclusion of this article.

2. Construction of numerical methods and their convergence analysis

This section discusses some well-known methods in the literature. One of the most ancient and popular numerical methods for approximating single roots of a nonlinear equation

$$f(x) = 0, \quad (2.1)$$

is Newton's method [26] (NM), which has local quadratic convergence.

$$x^{(t+1)} = x^{(t)} - \frac{f(x^{(t)})}{f'(x^{(t)})}, t = 0, 1, \dots \quad (2.2)$$

Kou et al. [27] presents an optimal 2nd-order iterative method (abbreviated as YO) as follows:

$$x^{(t+1)} = x^{(t)} - \left(1 + \frac{\left(\frac{f(x^{(t)})}{f'(x^{(t)})} \right)}{1 + (\alpha_1 + \alpha) \left(\frac{f(x^{(t)})}{f'(x^{(t)})} \right)} \right) \left(\frac{\left(\frac{f(x^{(t)})}{f'(x^{(t)})} \right)}{1 + \lambda_n \left(\frac{f(x^{(t)})}{f'(x^{(t)})} \right)} \right), \quad (2.3)$$

where $\lambda_n = \alpha = 1$ and $\alpha_1 = 0$.

Chicharro et al. [28] give the optimal local second-order iterative method (abbreviated as AC) as follows:

$$x^{(t+1)} = x^{(t)} - \left(\frac{f(x^{(t)})}{f'(x^{(t)}) + \alpha f(x^{(t)})} \right), t = 0, 1, \dots \quad (2.4)$$

We suggested the following optimal family of numerical iterative schemes abbreviated as BB1:

$$x^{(t+1)} = x^{(t)} - \left(\frac{f(x^{(t)})}{f'(x^{(t)})} \right) \left(\frac{1}{1 + \alpha \vartheta(u)} \right), t = 0, 1, \dots \quad (2.5)$$

where $u = \frac{f(x^{(t)})(f'(x^{(t)}))^\beta}{(f'(x^{(t)}))^2}$, $\beta, \alpha \in \mathbb{R}$.

The convergence order of iteration scheme Eq (2.5) is found by using CAS-Maple 18 in the form of the following theorem:

Theorem 1. Let I an open interval and $\zeta \in I$ be a simple root of a sufficiently differential function $f : I \subseteq \mathfrak{K} \rightarrow \mathfrak{K}$. For a real valued function $\vartheta(u)$ and $x^{(0)}$ sufficiently close to ζ with $\vartheta(0) = 0$, and $\vartheta'(0) < \infty$ then iterative scheme (2.5) has local quadratic convergence, satisfying the error equation:

$$e^{(t+1)} = (\alpha\vartheta'(0) + c_2)(e^{(t)})^2 + O((e^{(t)})^3), \quad (2.6)$$

where $c_m = \frac{f^{(m)}(\zeta)}{m!f'(\zeta)}$; $m \geq 2$.

Proof. Let $f(x)$ have a simple root ζ and $e^{(t)} = x^{(t)} - \zeta$. By a Taylor series expansion of $f(x^{(t)})$ around $x^{(t)} = \zeta$, taking $f(\zeta) = 0$, we get

$$f(x^{(t)}) = f'(\zeta)(e^{(t)}) + c_2(e^{(t)})^2 + c_3(e^{(t)})^3 + O((e^{(t)})^4), \quad (2.7)$$

and

$$f'(x^{(t)}) = f'(\zeta)(1 + 2c_2e^{(t)} + 3c_3(e^{(t)})^2 + 4c_4(e^{(t)})^3 + O((e^{(t)})^4)). \quad (2.8)$$

Dividing (2.7) by (2.8), we have

$$\frac{f(x^{(t)})}{f'(x^{(t)})} = e^{(t)} + c_2(e^{(t)})^2 + (2c_2^2 - 4c_3)(e^{(t)})^3 + O((e^{(t)})^4), \quad (2.9)$$

$$(f'(x^{(t)}))^2 = 1 + 4c_2e^{(t)} + (4c_2^2 + 6c_3)(e^{(t)})^2 + \dots \quad (2.10)$$

$$u = \frac{f(x^{(t)})(f'(x^{(t)}))^{\beta}}{(f'(x^{(t)}))^2} = e^{(t)}(6c_6(e^{(t)})^5 + 5c_5(e^{(t)})^4 + 4c_4(e^{(t)})^3 + \dots$$

Thus, using the Taylor series, we have

$$\vartheta(u) = \vartheta(0) + \vartheta'(0)u + \frac{\vartheta''(0)u^2}{2!} + \dots \quad (2.11)$$

$$\vartheta(u) = \vartheta(0) + (6c_6(e^{(t)})^5 + 5c_5(e^{(t)})^4 + 4c_4(e^{(t)})^3) + \dots \quad (2.12)$$

$$\frac{1}{1 + \alpha * \vartheta(u)} = \frac{1}{1 + \alpha\vartheta(0)} - \left(\frac{\tau\vartheta'(0)}{(\alpha\vartheta(0) + 1)^2} \right) e^{(t)} + O((e^{(t)})^2), \quad (2.13)$$

$$e^{(t+1)} = x^{(t)} - \zeta - \left(\frac{f(x^{(t)})}{f'(x^{(t)})} \right) \left(\frac{1}{1 - \alpha\vartheta(u)} \right) = \left(1 - \frac{1}{1 + \alpha\vartheta(0)} \right) e^{(t)} + \left(\frac{\tau\vartheta'(0)}{(1 + \alpha\vartheta(0))^2} + \frac{c_2}{1 + \alpha\vartheta(0)} \right) (e^{(t)})^2 + O((e^{(t)})^3), \quad (2.14)$$

and putting $\vartheta(0) = 0$ in (2.14), we have

$$e^{(t+1)} = (\alpha\vartheta'(0) + c_2)(e^{(t)})^2 + O((e^{(t)})^3). \quad (2.15)$$

This proves second-order convergence. □

Special cases:

Now we construct some special cases of the family of methods described by (2.5). We construct the following numerical schemes by choosing $\beta = 2$ in (2.5).

Method-1 (abbreviated as BB1): Take the function $\vartheta(u)$ defined as $\vartheta(u) = \frac{u}{1+u}$ which satisfies the condition of Theorem 1, i.e., $\vartheta'(0) < \infty$. Then, we have the following optimal family of iterative methods:

$$x^{(t+1)} = x^{(t)} - \left(\frac{f(x^{(t)})}{f'(x^{(t)})} \right) \left(\frac{1}{1 + \alpha \frac{f(x^{(t)})}{1+f(x^{(t)})}} \right). \quad (2.16)$$

Method-2 (abbreviated as BB2): Take the function $\vartheta(u)$ defined as $\vartheta(u) = \frac{u}{1+\gamma u^2}$ which satisfies the condition of Theorem 1, i.e., $\vartheta'(0) < \infty$. Then, we have the following optimal family of iterative methods:

$$x^{(t+1)} = x^{(t)} - \left(\frac{f(x^{(t)})}{f'(x^{(t)})} \right) \left(\frac{1}{1 + \alpha \left(\frac{f(x^{(t)})}{1+\gamma(f(x^{(t)})^2} \right)} \right). \quad (2.17)$$

Method-3 (abbreviated as BB3): Take the function $\vartheta(u)$ defined as $\vartheta(u) = \frac{u}{1+\gamma u}$ which satisfies the condition of Theorem 1, i.e., $\vartheta'(0) < \infty$. Then, we have the following optimal family of iterative methods:

$$x^{(t+1)} = x^{(t)} - \left(\frac{f(x^{(t)})}{f'(x^{(t)})} \right) \left(\frac{1}{1 + \alpha \left(\frac{f(x^{(t)})}{1+\gamma f(x^{(t)})} \right)} \right), \quad (2.18)$$

where $\alpha, \gamma \in \mathfrak{R}$.

2.1. Complex dynamical analysis of the family of numerical methods

This section discusses the stability of the numerical scheme Eq (2.16) against the backdrop of the complex dynamics. Iterative method Eq (2.16) leads to the following rational map:

$$\mathfrak{R}_f = x - \left(\frac{f(x)}{f'(x)} \right) \left(\frac{1}{1 + \alpha \frac{f(x)}{1+f(x)}} \right). \quad (2.19)$$

Recalling some basic concepts of this theory (detailed information can be found in [29–38]). Taking a rational function $\mathfrak{R}_f : \mathbb{C} \rightarrow \mathbb{C}$, \mathbb{C} denotes the Riemann sphere. The orbit $s_0 \in \mathbb{C}$ defines a set $orb(x) = \{x_0, \mathfrak{R}_f(x_0), \mathfrak{R}_f^2(x_0), \dots, \mathfrak{R}_f^m(x_0), \dots\}$. A point $x_0 \in \mathbb{C}$ is called a fixed point if $\mathfrak{R}_f(x_0) = x_0$. A particular fixed point x_0 is called a strange fixed point if $f(x^r) = 0$ when $x_0 \neq x^r$. A T-periodic point is defined as the point $x_T \in \mathbb{C}$ satisfying $\mathfrak{R}^T(x_T) = s_T$ with $\mathfrak{R}'(x_T) \neq x_T$ for $x < T$. If x_0 is a fixed point of \mathfrak{R} , then it is:

- Super attracting if $|\mathfrak{R}'_f(x_0)| = 0$.
- Attracting if $|\mathfrak{R}'_f(x_0)| < 1$.
- Repulsive if $|\mathfrak{R}'_f(x_0)| > 1$.

- Neutral if $|\mathfrak{R}'_f(x_0)| = 1$.
- A strange fixed point if it not associated to any root of non-linear equation (2.1).

An attracting point $x^* \in \mathbb{C}$ defines a basin of attraction $\mathfrak{R}(x^*)$ as the set of starting points whose orbit tends to x^* .

The iterative method BB1 holds the scaling theorem which allows a suitable change of co-ordinate to reduce of dynamics of iteration of general maps to the study of a specific family of iteration of similar maps. The one point iterative method (2.16) has a universal Julia set if a rational map exist which conjugates by the Mobius transformation.

Theorem 2. A rational map $f_g(x)$ arising from (2.16) applied to $f(x) = (x - a)(x - b)$, where $a, b \in \mathfrak{R}$, $f_g(x)$ is conjugate via the Mobius transformation by $M(v) = \frac{v-a}{v-b}$ to

$$f_g(x, \alpha) = \frac{x^2(4\alpha + x + 1)}{(4\alpha + 1)x + 1}.$$

Proof. Let $f(x) = (x - a)(x - b)$, where $a, b \in \mathfrak{R}$. The Möbius transformation given by:

$$M(v) = \frac{v - a}{v - b}, \quad (2.20)$$

with inverse

$$[M(v)]^{-1} = \frac{vb - a}{v - 1}, \quad (2.21)$$

which we consider as a map from $\mathbb{C} \cup \infty$.

Then, we have

$$f_g(x, \alpha) = \frac{x^2(4\alpha + x + 1)}{(4\alpha + 1)x + 1}. \quad (2.22)$$

□

Similarly, we can get the following conclusions.

Theorem 3. A map $f_g(x)$ (rational) arising from BB2, BB3, AC, NM, and YO applied to $f(x) = (x - a)(x - b)$, where $a, b \in \mathfrak{R}$, $f_g(x)$ is conjugate via the Mobius transformation by $M(v) = \frac{v-a}{v-b}$ to

$$f_g(x, \alpha) = \frac{x^2(4\alpha x^3 + s^4 - 4\alpha x^2 - 4x^3 - 4\alpha x - 2x^2 + 4\alpha - 4x + 1)}{(4\alpha + 1)x^4 - 4(\alpha + 1)x^3 - 2(2\alpha + 1)x^2 + 4x(\alpha - 1) + 1}, \quad (2.23)$$

$$f_g(x, \alpha) = \frac{x^2(s^2 + (4\alpha - 6.2)x + 1 + 4\alpha)}{(1 + 4\alpha)x^2 + (4\alpha - 6.2)x + 1}, \quad (2.24)$$

$$f_g(x, \alpha) = \frac{x^2(2\alpha - x + 1)}{(2\alpha - 1)x + 1}, \quad (2.25)$$

$$f_g(x, \alpha) = x^2, \quad (2.26)$$

$$f_g(x, \alpha) = \frac{x^2(4\alpha^2 x - 4\alpha x^2 + x^3 + 2\alpha x - x^2 + 2\alpha + x - 1)}{(4\alpha^2 - 2\alpha)x^2 - 2\alpha x^3 + 3x^3 + 4\alpha x - 3x^2 - x + 1}. \quad (2.27)$$

The fixed points of (2.22) are $x = -1$, $x = 0$, $x = 1$, and $x = \infty$. We calculate $f'_g(x, \alpha)$ for the stability of fixed points of BB1 as:

$$f'_g(x, \alpha) = \frac{2x((4\alpha + 1)x^2 + (8\alpha^2 + 4\alpha + 2)x + (4\alpha + 1))}{((4\alpha + 1)x + 1)^2}. \quad (2.28)$$

From (2.28) it is clear that $x = 0$ and ∞ are always super attractive fixed points but the stability of other fixed points are depends on α , which we present here. The operator $f'_g(x, \alpha)$ for $x = -1$ gives

$$|f'_g(-1, \alpha)| = \left| \frac{2\alpha - 1}{\alpha} \right|. \quad (2.29)$$

Analyzing (2.29), as $\alpha \rightarrow \pm\infty$ we obtain horizontal asymptotes for $|f'_g(-1, \alpha)| = 1$ and vertical asymptotes for $\alpha = 0$ (Newton method).

Theorem 4. *The strange point $x = -1$ is known as:*

- (i) If $\Re_e(\alpha) < \frac{1}{4}$, then $x = -1$ is an attractor and it can be super attractor for $\alpha = -1$.
- (ii) When $\Re_e(\alpha) = \frac{1}{4}$, is a parabolic point for $x = -1$.
- (iii) If $\alpha \neq 0$ and $\Re_e(\alpha) > \frac{1}{4}$, is a repulsive for $x = -1$.

Proof. As

$$f'_g(-1, \alpha) = \frac{2\alpha - 1}{\alpha}, \quad (2.30)$$

then,

$$\left| \frac{2\alpha - 1}{\alpha} \right| \leq 1 \text{ is equivalent to } |2\alpha - 1| \leq |2\alpha|. \quad (2.31)$$

Let us consider $\alpha = a + ib \in \mathbb{C}$.

Then,

$$\begin{aligned} -4a + 1 &\leq 0, \\ a &\leq \frac{1}{4}. \end{aligned} \quad (2.32)$$

Thus,

$$|f'_g(1, \alpha)| \leq 1 \text{ iff } a \leq \frac{1}{4}. \quad (2.33)$$

Finally, if $a \neq 0$ and $\Re_e(\alpha) > \frac{1}{4}$, then $|f'_g(-1, \alpha)| \geq 1$. □

The operator $f'_g(1, \alpha)$ gives

$$|f'_g(1, \alpha)| = \left| \frac{2(\alpha + 1)}{2\alpha + 1} \right|. \quad (2.34)$$

By analyzing (2.34), as $\alpha \rightarrow \pm\infty$ we obtain horizontal asymptotes in $|f'_g(1, \alpha)| = 1$.

Theorem 5. *The character of the strange point $x = 1$ is as follows:*

- (i) If $\Re_e(\alpha) > -\frac{3}{4}$, it is an attractor for $x = 1$.
- (ii) When $\Re_e(\alpha) = -\frac{3}{4}$, it is a parabolic point for $x = 1$.
- (iii) If $\alpha \neq 0$ and $\Re_e(\alpha) < -\frac{3}{4}$, then $x = 1$ a repulsive.

Proof. As

$$f'_g(1, \alpha) = \frac{2(\alpha + 1)}{2\alpha + 1}, \quad (2.35)$$

then,

$$\left| \frac{2(\alpha + 1)}{2\alpha + 1} \right| \leq 1 \text{ is equivalent to } |2(\alpha + 1)| \leq |2\alpha + 1|. \quad (2.36)$$

Consider $\alpha = a + ib \in \mathbb{C}$.

Then,

$$4\alpha + 3 \leq 0. \quad (2.37)$$

Thus,

$$|f'_g(1, \alpha)| \leq 1 \text{ iff } a \leq -\frac{3}{4}. \quad (2.38)$$

Finally, if $a \neq 1$ and $\Re_e(\alpha) > -\frac{3}{4}$, then $|f'_g(1, \alpha)| < 1$ □

The stability functions of the iterative schemes BB1 are as follows:

$$\{f'_g(-1, \alpha) = \min\{f'_g(-1, \alpha), -1\}, f'_g(1, \alpha) = \min\{f'_g(1, \alpha), 1\}\}. \quad (2.39)$$

Figure 1 represents the region of stability of strange fixed points. Left: Figure $f'_g(1, \alpha)$. Right: Figure $f'_g(-1, \alpha)$.

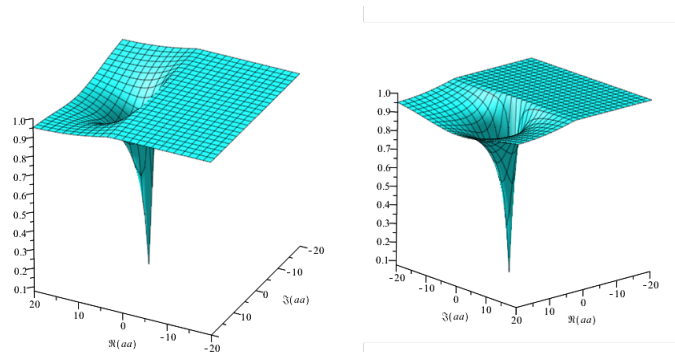


Figure 1. Zone of stability of strange fixed points of numerical scheme BB1. Left-zone of stability of strange fixed point $x = -1$; Right-zone of stability of strange fixed point $x = 1$.

Analysis of Critical Points:

The critical points of BB1 satisfy $f'_g(-1, \alpha) = 0$ i.e., $x = 0, x = \infty$ and

$$\begin{cases} ccr1 = \frac{-4\alpha^2 + 2\sqrt{4\alpha^4 + 4\alpha^3 - \alpha^2 - \alpha} - 2\alpha - 1}{4\alpha + 1}, \\ ccr2 = -\frac{4\alpha^2 + 2\sqrt{4\alpha^4 + 4\alpha^3 - \alpha^2 - \alpha} + 2\alpha + 1}{4\alpha + 1}. \end{cases} \quad (2.40)$$

If $\alpha \neq 1, 1/4$, then observe that $cr1 = \frac{1}{cr2}$ and $cr1 = cr2 = 1$ only when $\alpha = -1$, while $cr1 = cr2 = -1$ only when $\alpha = 0$. Figure 2 presents the behavior of the fixed and critical points for real of α . Fixed points $-1, 0$, and 1 are represented by orange, black, and green dotted lines, while critical points $(cr1, cr2)$ are represented by red and blue lines, respectively.

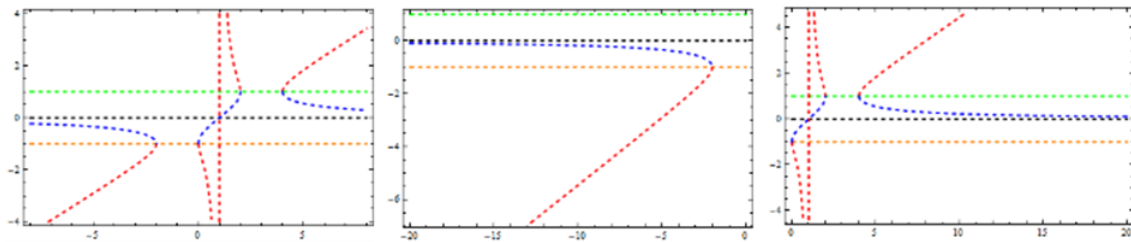


Figure 2. Dynamical behavior of all strange fixed and critical point of numerical scheme BB1. Dynamical behavior of strange fixed points and critical points for $0 \leq \alpha \leq 20$ (left), $-20 \leq \alpha \leq 0$ (middle), $-10 \leq \alpha \leq 10$ (right).

Figure 3: The union of respective stability function of the all strange fixed points of BB1-BB3, AC, and YO, respectively, from left to right.

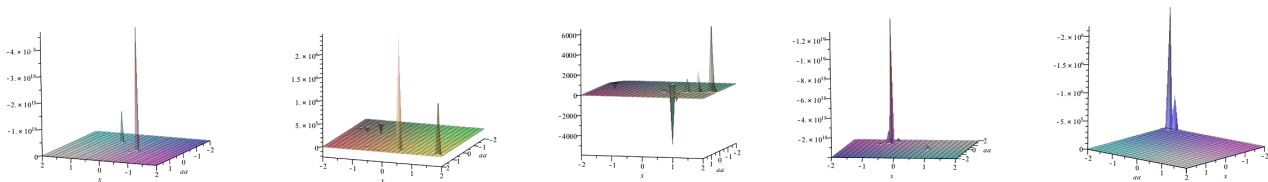


Figure 3. The union of the respective stability function of the all strange fixed points of BB1–BB3, AC, and, YO respectively.

Theorem 6. *The only member of BB1 whose operator is always conjugated to the rational map x^2 is the element corresponding to $\alpha = 0$.*

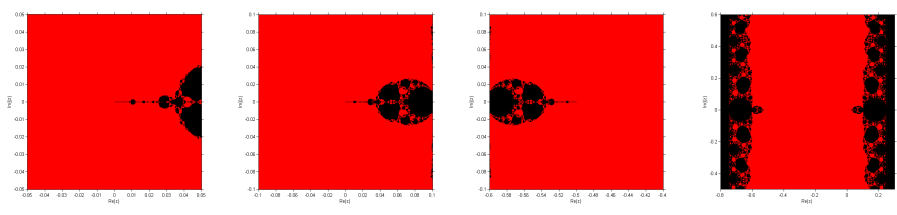
Proof. From (2.22), we represent $nn(x) = x^2((4\alpha + x + 1))$ in the numerator and $dd(x) = (4\alpha + 1)x + 1$ denominator of (2.22). Then, for $\alpha = 0$, we get $nn(x) = dd(x)$. \square

Parametric planes:

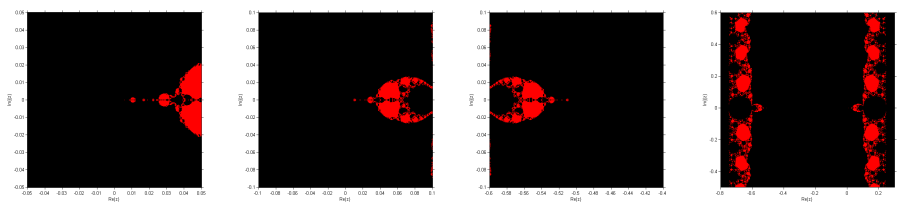
We generate a parametric plane by taking α over a mesh of 250,000 points in a complex plane. Critical points are chosen as initial guessed values and then iterate until it converges to any fixed points. Taking 10^{-3} as a tolerance. We paint the complex parameter value in red if it converges, and black in other cases. The parametric planes for the critical points are shown in Figure 4.

Dynamical planes:

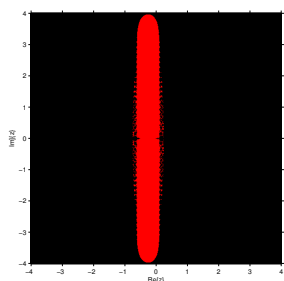
Dynamical planes are generated as parametric planes. To generate them, the real and imaginary parts of the starting estimations are represented as two axes over a mesh of 250×250 , and the same stopping criteria as in the parametric plane. The study assigns different colors to indicate which root the method converges to, and black in other cases. The dynamical planes of the iterative scheme BB1, YO, NM, and AC for different parametric values are shown in Figures 5–7.



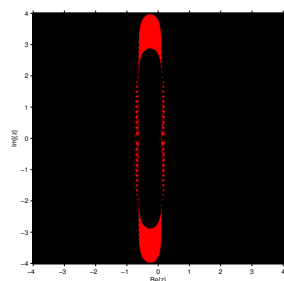
(a) Details of the parametric plane for the critical points cr1.



(b) Details of the parametric plane for the critical points cr2.



(c) Parametric planes correspond to the critical point cr1.



(d) Parametric planes correspond to the critical point cr2.

Figure 4. (a–d): The parametric planes of the numerical scheme BB1 correspond to the critical points cr1 and cr2.

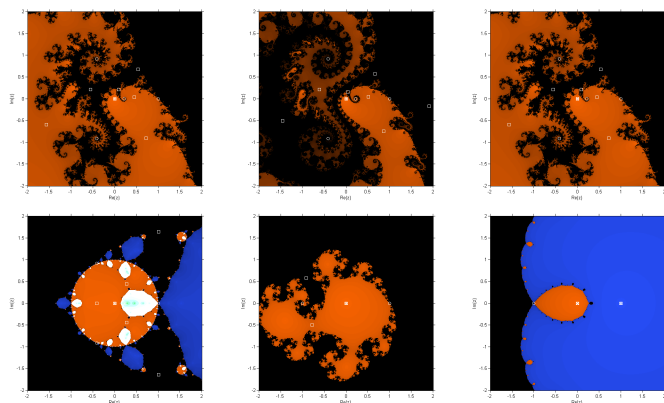


Figure 5. Iterative algorithms BB1 with dynamical planes are demonstrated from top left to bottom right for various values of the parameter $\alpha = -2i, 2i, 3i, -1, 0.1 - 0.1i, 1$.

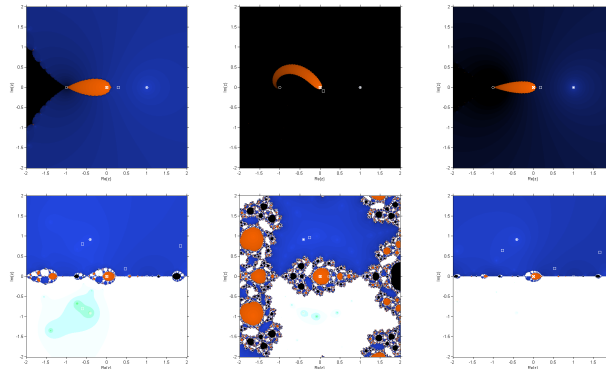


Figure 6. Iterative algorithms BB1 with dynamical planes are demonstrated from top left to bottom right for various values of the parameter $\alpha = -2, -3 - 3i, -3, 1, 2, 3$.

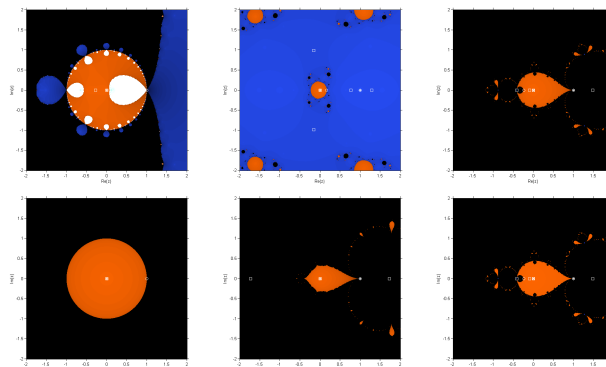


Figure 7. Iterative algorithms (BB1–BB3, AC, YO, NM) with dynamical planes are demonstrated from top left to bottom right end for various values of the parameter.

In Table 1, Els-Time denotes the amount of time in seconds that has passed, Start-Points denotes the number of starting points (62500.00 in a square), Average-It denotes the average number of iterations, C-Points denotes the number of converging points, and D-Points denotes the number of diverging points utilized to form dynamical planes (the basins of attractions [39, 40]). Table 1 clearly shows that our recently developed methods BB1–BB3 outperform the existing iterative methods in terms of CPU-Time, Average-It, T-Points, C-Points, and D-Points.

In Table 2, Els-Time denotes the amount of time in seconds that has passed, Start-Points denotes the number of starting points (62500.00 in a square), Average-It denotes the average number of iterations, C-Points denotes the number of converging points, and D-Points denotes the number of diverging points utilized to form dynamical planes (the basins of attractions). Table 2 clearly shows that our recently developed methods BB1–BB3 outperform existing iterative methods in terms of CPU-Time, Average-It, T-Points, C-Points, and D-Points.

In Table 3, Els-Time denotes the amount of time in seconds that has passed, Start-Points denotes the number of starting points (62500.00 in a square), Average-It denotes the average number of iterations, C-Points denotes the number of converging points, and D-Points denotes the number of diverging points utilized to form dynamical planes (the basins of attractions). Table 3 clearly shows that our

recently developed methods BB1-BB3 outperform the existing iterative methods in terms of CPU-Time, Average-It, T-Points, C-Points, and D-Points.

Table 1. Convergence divergence analysis of numerical schemes for $f_1(x)$.

Method	BB1	BB2	BB3	AC	YO	NM
Els-Time	0.0124000	0.02415	0.01584	0.214501	0.021450	0.0456211
Average-It	3.4500000	3.50000	3.00000	5.100000	5.200000	7.4000000
T-Points	62500.000	62500.00	62500.00	62500.00	62500.00	62500.000
D-Points	340.00000	105.3500	458.3500	214.4500	1541.450	765.45100
C-Points	62160.000	62394.65	62041.65	62285.55	60958.54	61734.549

Table 2. Convergence divergence analysis of numerical schemes for $f_2(x)$.

Method	BB1	BB2	BB3	AC	YO	NM
Els-Time	0.002100	0.004350	0.012140	0.0112501	0.011471	0.012671
Average-It	4.450124	5.504120	6.704512	8.102510	9.201200	10.40010
T-Points	62500.00	62500.00	62500.00	62500.00	62500.00	62500.00
D-Points	210.0000	10.35000	115.3500	414.4500	80.45000	105.4510
C-Points	62290.00	62489.65	62384.65	6205.550	62419.55	62394.549

Table 3. Convergence divergence analysis of numerical schemes for $f_3(x)$.

Method	BB1	BB2	BB3	AC	YO	NM
Els-Time	0.0324	0.03412	0.03481	0.060121	0.071010	0.0557001
Average-It	5.0500	4.5000	4.0000	5.6000	4.2000	8.1000
T-Points	62500.00	62500.00	62500.00	62500.00	62500.00	62500.00
D-Points	1940.000	345.3010	298.6700	451.8700	984.5700	765.4510
C-Points	60560.00	62154.699	62201.33	62048.13	61515.43	61734.549

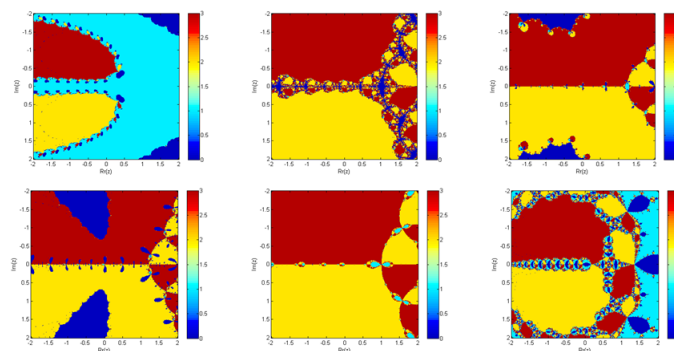


Figure 8. The basin of attraction for the numerical schemes BB1–BB3, AC, YO, and NM technique, respectively, is shown from left to right (for polynomial equation used in example 1).

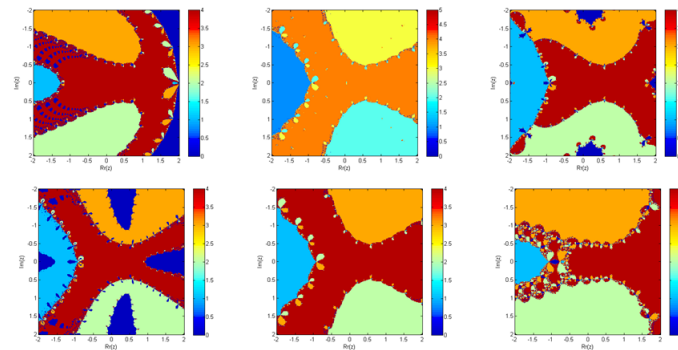


Figure 9. The basin of attraction for the numerical schemes BB1–BB3, AC, YO, and NM technique, respectively, is shown from left to right (for polynomial equation used in example 2).

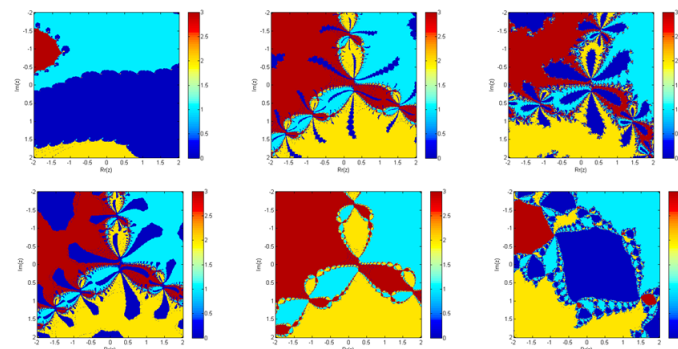


Figure 10. The basin of attraction for the numerical schemes BB1–BB3, AC, YO, and NM technique, respectively, is shown from left to right (for polynomial equation used in example 3).

2.2. Engineering applications

This section discusses an engineering application to check the efficiency and performance of the newly developed methods compared to existing methods, i.e., the AC, YO, and NM methods.

Example 1. Current-Voltage Diode [41]

A manufacturing company provided a tunnel diode with voltage-current output:

$$I = I(x) = x^3 - 1.5x^2 + 0.6x.$$

The tunnel diode is connected with resistor R and voltage source E . By applying Kirchhoff's voltage law, we find a relation for the steady current as follows:

$$I(x) = \frac{E - x}{R}.$$

For a given $E=1.4$ and $R=19$, we find x by setting $f(x)=0$,

$$\frac{14 - x}{19} = x^3 - 1.5x^2 + 0.6x,$$

$$f_1(x) = 19x^3 - 28.5x^2 + 12.4x - 1.4. \quad (2.41)$$

The exact roots of (2.41) are

$$\zeta_1 = 0.175236, \zeta_2 = 0.527234, \zeta_3 = 0.797529.$$

The initial estimates of $f_1(x)$ have been taken as

$$x_1^{(0)} = 0.1, x_2^{(0)} = 0.5, x_3^{(0)} = 0.7.$$

Table 4 clearly shows the dominance behavior of BB1–BB3 over the AC, YO, and NM iterative methods in approximated absolute error on various number of iterations n for the non-linear function $f_1(x)$. Various initial guess values are used in order to approximate all roots of $f_1(x)$.

Table 4. Simple roots for polynomial equations $f_1(x)$ used in engineering application 1.

Method	n	$f(x^{(t)})$	$x^{(t)} - x^{(t-1)}$	cpu	LCOC	Ex-R
Numerical results of the iterative methods by taking $x^{(0)} = 0.2$ to approximate of the first root						
NM	06	7.4e-20	1.0e-37	0.0331454	1.91479	0.125236273796841722315
YO	06	9.4e-20	1.6e-37	0.0241564	1.98748	0.125236273796841722315
AC	06	3.8e-21	2.7e-38	0.0145187	2.01455	0.125236273796841722315
BB1-BB3	06	4.2e-26	3.3e-41	0.0112475	2.11748	0.125236273796841722315
Numerical results of the iterative methods by taking $x^{(0)} = 0.4$ to approximate of the first root						
NM	06	1.1e-60	2.0e-120	0.0456154	2.000000	0.527234489565474586149
YO	06	3.3e-65	4.3e-119	0.0345136	2.000001	0.527234489565474586149
AC	06	1.7e-64	4.0e-128	0.0215465	2.000005	0.527234489565474586149
BB1-BB3	06	4.1e-67	2.6e-135	0.0125487	2.000000	0.527234489565474586149
Numerical results of the iterative methods by taking $x^{(0)} = 0.8$ to approximate of the first root						
NM	08	3.0e-05	1.6e-07	0.0641568	1.765489	0.797529236845714584845
YO	07	5.0e-07	4.3e-12	0.0545147	1.894651	0.797529236845714584845
AC	06	5.2e-04	4.6e-06	0.2451349	1.874563	0.797529236845714584845
BB1-BB3	06	3.1e-11	2.7e-31	0.1413254	1.998745	0.797529236845714584845

Example 2. Mass-Spring system [42]

A spring develops a non-linear restoring force given by

$$F = r_1x + r_3x^3 + r_5x^5, \quad (2.42)$$

where F represents force in kg force and x is displacement in cm. Using value of constants $r_1 = 5.25\text{kg/cm}$, $r_3 = 0.60\text{kg/cm}$, $r_5 = 0.0118\text{kg/cm}$ in (2.42), we get a non-linear equation in terms of the displacement x

$$f_2(x) = 0.0118x^5 + 0.6x^3 + 5.25x. \quad (2.43)$$

The exact roots of (2.43) are

$$\zeta_1 = 0, \zeta_2 = -3.35118i, \zeta_3 = 3.35118i, \zeta_4 = -6.29421i, \zeta_5 = 6.29421i.$$

The initial estimates for simultaneous determination of all roots of non-linear function $f_2(x)$ has been taken as:

$$x_1^{(0)} = 0.1, x_2^{(0)} = -3.3i, x_3^{(0)} = 3.3i, x_4^{(0)} = -6.2i, x_5^{(0)} = 6.2i.$$

Table 5 clearly shows the dominance behavior of BB1–BB3 over the AC, YO, and NM iterative methods in approximated absolute error on various number of iterations n for the non-linear function $f_2(x)$. Various initial guess values are used in order to approximate all roots of $f_2(x)$.

Table 5. Simple roots for polynomial equations $f_2(x)$ used in engineering application 2.

Method	n	$f(x^{(t)})$	$x^{(t)} - x^{(t-1)}$	CPU-time	LCOC	Ex-R
Numerical results of the iterative methods by taking $x^{(0)} = 0.3$ to approximate of the first root						
NM	06	0.4e-27	1.0e-47	0.0331454	1.714794	0.0000000000000000000001
YO	06	4.5e-25	0.6e-59	0.0241564	2.087485	0.0000000000000000000001
AC	06	3.0e-35	2.7e-68	0.0145187	1.914551	0.0000000000000000000001
BB1-BB3	06	4.1e-38	0.3e-75	0.0112475	2.317485	0.0000000000000000000001

Example 3. Fractional conversion of magnesium hydroxide in hydrochloric acid: [43–46].

The acidity of a saturated solution of magnesium hydroxide $MgOH$ in hydrochloric acid HCl is given by

$$\frac{3.64 \times 10^{-11}}{[H_3O^+]} = [H_3O^+] + 3.6 \times 10^{-4}, \quad (2.44)$$

for the hydronium ion concentration $[H_3O^+]$. If we set $x = 10^4 [H_3O^+]$, we obtain the following non-linear equation

$$f_3(x) = x^3 + 3.6x^2 - 36.4, \quad (2.45)$$

with exact roots $2.4, -3.0 \pm 2.3i$ up-to one decimal place. The initial estimates of (2.45) have been taken as:

$$x_1^{(0)} = 2.45, x_2^{(0)} = -3.0261 + 2.3834i, x_3^{(0)} = -3.0261 - 2.3834i.$$

Table 6 clearly shows the dominance behavior of BB1–BB3 over the AC, YO, and NM iterative methods in approximated absolute error on various number of iterations n for the non-linear function $f_3(x)$. Various initial guess values are used in order to approximate all roots of $f_3(x)$.

Table 6. Simple roots for polynomial equations $f_3(x)$ used in engineering application 3.

Method	n	$f(x^{(t)})$	$x^{(t)} - x^{(t-1)}$	cpu	LCOC	Ex-R
Numerical results of the iterative methods by taking $x^{(0)} = 0.2$ to approximate of the first root						
NM	07	0.9e-25	1.7e-47	0.0331454	1.91479	2.95361891939385560115508
YO	06	7.7e-27	1.6e-47	0.0241564	1.98748	2.95361891939385560115508
AC	06	5.1e-33	8.1e-57	0.0145187	2.01455	2.95361891939385560115508
BB1-BB3	06	9.2e-37	9.3e-64	0.0112475	2.11748	2.95361891939385560115508

3. Conclusions

We developed a family of iterative techniques with two parameters for locating the roots of nonlinear equations. Due to the lengthy and complicated mathematical formulas used in the convergence study, we used symbolic computation utilizing several programs created in the Mathematica computer system. We implemented a different method to evaluate the root solver efficiency utilizing parametric planes (Figures 1–4) and dynamical planes (Figures 5–7) to select the

best parameter value. The stability region and dynamical behavior of the strange fixed points were discussed. Root-finding methods can be analyzed graphically using this approach concerning various starting points. Well-built software will offer a visual knowledge of the convergence behavior of iterative methods for establishing comparison standards to show the basins of attraction (Figures 8–10). However, it will also consider various qualitative factors, such as CPU time and the convergence region. The numerical results show that the newly developed technique outperforms the current method regarding the residual error, CPU-time, and iterations. In the future, we will construct higher-order efficient, optimal, and stable iterative methods for finding simple as well as all distinct and multiple roots of Eq (2.1). We may explore the applications of these results in other areas. For example, the dynamics nonlinear inventory management system, numerical solution of fractional SIR epidemiological model, stability and optimal control strategies for a novel epidemic model of COVID-19.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. M. Cosnard, P. Fraigniaud, Finding the roots of a polynomial on an MIMD multicomputer, *Parall. Comput.*, **15** (1990), 75–85. [https://doi.org/10.1016/0167-8191\(90\)90032-5](https://doi.org/10.1016/0167-8191(90)90032-5)
2. X. Lü, H. Hui, F. Liu, Y. Bai, Stability and optimal control strategies for a novel epidemic model of COVID-19, *Nonlinear Dyn.*, **106** (2021), 1491–1507. <https://doi.org/10.1007/s11071-021-06524-x>
3. A. Naseem, M. Rehman, T. Abdeljawad, Computational methods for non-linear equations with some real-world applications and their graphical analysis, *Intell. Autom. Soft Comput.*, **30** (2021), 805–819. <http://dx.doi.org/10.32604/iasc.2021.019164>
4. B. Liu, X. E. Zhang, B. Wang, X. Lü, Rogue waves based on the coupled nonlinear Schrödinger option pricing model with external potential, *Mod. Phys. Lett. B*, **36** (2022), 2250057. <https://doi.org/10.1142/S0217984922500579>
5. O. Aberth, Iteration methods for finding all zeros of a polynomial simultaneously, *Math. Comput.*, **27** (1973), 339–344.
6. S. M. Kang, A. Rafiq, S. Ahmad, Y. C. Kwun, New iterative method with higher-order convergence for scalar equations, *Int. J. Math. Anal.*, **10** (2016), 339–356. <http://dx.doi.org/10.12988/ijma.2016.612>
7. O. S. Solaiman, I. Hashim, Optimal eighth-order solver for nonlinear equations with applications in chemical engineering, *Intell. Autom. Soft Comput.*, **27** (2021), 379–390. <http://dx.doi.org/10.32604/iasc.2021.015285>

8. H. T. Kung, J. F. Traub, Optimal order of one-point and multipoint iteration, *J. ACM*, **21** (1974), 643–651. <https://doi.org/10.1145/321850.321860>
9. M. Shams, N. Rafiq, N. Kausar, P. Agarwal, C. Park, S. Momani, Efficient iterative methods for finding simultaneously all the multiple roots of polynomial equation, *Adv. Differ. Equ.*, **2021** (2021), 495. <https://doi.org/10.1186/s13662-021-03649-6>
10. S. A. Sariman, I. Hashim, New optimal Newton-Householder methods for solving nonlinear equations and their dynamics, *Comput. Mater. Contin.*, **65** (2020), 69–85. <http://dx.doi.org/10.32604/cmc.2020.010836>
11. M. A. Noor, K. I. Noor, W. A. Khan, F. Ahmad, On iterative methods for nonlinear equations, *Appl. Math. Comput.*, **183** (2006), 128–133. <http://dx.doi.org/10.1016/j.amc.2006.05.054>
12. O. Bazighifan, P. Kumam, Oscillation theorems for advanced differential equations with p-Laplacian like operators, *Mathematics*, **8** (2020), 821. <https://doi.org/10.3390/math8050821>
13. O. Moaaz, R. A. El-Nabulsi, O. Bazighifan, Oscillatory behavior of fourth-order differential equations with neutral delay, *Symmetry*, **12** (2020), 371. <https://doi.org/10.3390/sym12030371>
14. O. Bazighifan, H. Alotaibi, A. A. A. Mousa, Neutral delay differential equations: Oscillation conditions for the solutions, *Symmetry*, **13** (2021), 101. <https://doi.org/10.3390/sym13010101>
15. R. A. El-Nabulsi, O. Moaaz, O. Bazighifan, New results for oscillatory behavior of fourth-order differential equations, *Symmetry*, **12** (2020), 136. <https://doi.org/10.3390/sym12010136>
16. O. Moaaz, D. Chalishajar, O. Bazighifan, Some qualitative behavior of solutions of general class of difference equations, *Mathematics*, **7** (2019), 585. <https://doi.org/10.3390/math7070585>
17. D. A. Knoll, E. K. David, Jacobian-free Newton-Krylova methods: A survey of approaches and applications, *J. Comput. Phys.*, **193** (2004), 357–397. <https://doi.org/10.1016/j.jcp.2003.08.010>
18. J. F. Lemieux, B. Tremblay, J. Sedláček, P. Tupper, S. Thomas, D. Huard, et al., Improving the numerical convergence of viscous-plastic sea ice models with the Jacobian-free Newton-Krylova method, *J. Comput. Phys.*, **229** (2010), 2840–2852. <https://doi.org/10.1016/j.jcp.2009.12.011>
19. X. Y. Wu, A new continuation Newton-like method and its deformation, *Appl. Math. Comput.*, **112** (2000), 75–78. [https://doi.org/10.1016/S0096-3003\(99\)00049-1](https://doi.org/10.1016/S0096-3003(99)00049-1)
20. U. K. Qureshi, Z. A. Kalhor, A. A. Shaikh, A. R. Nangraj, Trapezoidal second order convergence method for solving nonlinear problems, *USJICT*, **2** (2018), 111–114.
21. J. H. He, Variational iteration method-some recent results and new interpretations, *J. Comput. Appl. Math.*, **207** (2007), 3–17. <https://doi.org/10.1016/j.cam.2006.07.009>
22. B. Saheya, G. Q. Chen, Y. K. Sui, C. Y. Wu, A new Newton-like method for solving nonlinear equations, *SpringerPlus*, **5** (2016), 1269. <https://doi.org/10.1186/s40064-016-2909-7>
23. S. Abbasbandy, Improving Newton-Raphson method for nonlinear equations by modified Adomian de composition method, *Appl. Math. Comput.*, **145** (2003), 887–893. [https://doi.org/10.1016/S0096-3003\(03\)00282-0](https://doi.org/10.1016/S0096-3003(03)00282-0)
24. G. M. Sandquist, Z. R. Wilde, *Introduction to System Science with MATLAB*, Hoboken: John Wiley and Sons, 2023.

25. A. Sohail, Genetic algorithms in the fields of artificial intelligence and data sciences, *Ann. Data Sci.*, **10** (2023), 1007–1018. <https://doi.org/10.1007/s40745-021-00354-9>
26. A. Y. Özban, Some new variants of Newton's method, *Appl. Math. Lett.*, **17** (2004), 677–682. [https://doi.org/10.1016/S0893-9659\(04\)90104-8](https://doi.org/10.1016/S0893-9659(04)90104-8)
27. J. Kou, Y. Li, A family of new Newton-like method, *Appl. Math. Comput.*, **192** (2007), 162–167. <https://doi.org/10.1016/j.amc.2007.02.129>
28. F. I. Chicharro, A. Cordero, N. Garrido, J. R. Torregrosa, Generating root-finder iterative methods of second order convergence and stability, *Axioms*, **8** (2019), 55. <https://doi.org/10.3390/axioms8020055>
29. A. Cordero, J. R. Torregrosa, Variants of Newton's method using fifth order quadrature formulas, *Appl. Math. Comput.*, **190** (2007), 686–698. <https://doi.org/10.1016/j.amc.2007.01.062>
30. J. Gerlach, Accelerated convergence in Newton's method, *SIAM Review*, **36** (1994), 272–276. <https://doi.org/10.1137/1036057>
31. I. Gościński, K. Gdawiec, Visual analysis of dynamics behavior of an iterative method depending on selected parameters and modifications, *Entropy*, **1** (2020), 737. <https://doi.org/10.3390/e22070734>
32. S. A. Siddiqui, A. Ahmad, Implementation of Newton's algorithm using FORTRAN, *SN Comput. Sci.*, **1** (2020), 348. <https://doi.org/10.1007/s42979-020-00360-3>
33. M. Shams, N. Rafiq, N. Kausar, N. A. Mir, A. Alalyani, Computer oriented numerical scheme for solving engineering problems, *Comput. Syst. Sci. Eng.*, **42** (2022), 689–701. <http://dx.doi.org/10.32604/csse.2022.022269>
34. S. Amat, S. Busquier, S. Plaza, Chaotic dynamics of a third-order Newton-type method, *J. Math. Anal. Appl.*, **336** (2010), 24–32. <https://doi.org/10.1016/j.jmaa.2010.01.047>
35. P. Fatou, Sur les equations fonctionnelles, *Bull. Soc. Mat. France*, **47** (1919), 161–271.
36. J. F. Traub, Computational complexity of iterative processes, *SIAM J. Comput.*, **1** (1972), 167–179. <https://doi.org/10.1137/0201012>
37. S. Kumar, J. Bhagwan, L. Jäntschi, Numerical simulation of multiple roots of van der Waals and CSTR problems with a derivative-free technique, *AIMS Math.*, **8** (2023), 14288–14299. <http://dx.doi.org/10.3934/math.2023731>
38. G. Thangkhenpau, S. Panday, L. C. Bolundut, L. Jäntschi, Efficient families of multi-point iterative methods and their self-acceleration with memory for solving nonlinear equations, *Symmetry*, **15** (2023), 1546. <https://doi.org/10.3390/sym15081546>
39. E. Sharma, S. Panday, S. K. Mittal, D. M. Joita, L. L. Pruteanu, L. Jäntschi, Derivative-free families of with-and without-memory iterative methods for solving nonlinear equations and their engineering applications, *Mathematics*, **11** (2023), 4512. <https://doi.org/10.3390/math11214512>
40. S. Kumar, J. R. Sharma, J. Bhagwan, L. Jäntschi, Numerical solution of nonlinear problems with multiple roots using derivative-free algorithms, *Symmetry*, **15** (2023), 1249. <https://doi.org/10.3390/sym15061249>
41. R. L. Burden, J. D. Faires, Boundary-value problems for ordinary differential equations, In: *Numerical Analysis*, 9th edition, Boston: Belmont Thomson Brooks/Cole Press, 2011.

42. V. K. Srivastav, S. Thota, M. Kumar, A new trigonometrical algorithm for computing real root of non-linear transcendental equations, *Int. J. Appl. Comput. Math.*, **5** (2019), 44. <https://doi.org/10.1007/s40819-019-0600-8>
43. M. I. Argyros, I. K. Argyros, S. Regmi, S. George, Generalized three step numerical methods for solving equations in Banach spaces, *Mathematics*, **10** (2022), 2621. <https://doi.org/10.3390/math10152621>
44. T. Lei, R. M. Y. Li, H. Fu, Dynamics analysis and fractional-order approximate entropy of inventory management nonlinear systems, *Math. Prob. Eng.*, **2021** (2021), 5516703. <https://doi.org/10.1155/2021/5516703>
45. W. Li, L. Weng, K. Zhao, S. Zhao, P. Zhang, Research on the evaluation of real estate inventory management in China, *Land*, **10** (2021), 1283. <https://doi.org/10.3390/land10121283>
46. H. Ran, Construction and optimization of inventory management system via cloud-edge collaborative computing in supply chain environment in the Internet of Things era, *Plos One*, **16** (2021), e0259284. <https://doi.org/10.1371/journal.pone.0259284>



AIMS Press

© 2024 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)