

Received January 15, 2022, accepted February 7, 2022, date of publication February 10, 2022, date of current version February 24, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3150775

A Novel Root-Finding Algorithm With Engineering Applications and Its Dynamics via Computer Technology

AMIR NASEEM¹, M. A. REHMAN¹, AND THABET ABDELJAWAD^{2,3}

¹Department of Mathematics, University of Management and Technology, Lahore 54770, Pakistan

²Department of Mathematics and Sciences, Prince Sultan University, Riyadh 11586, Saudi Arabia

³Department of Medical Research, China Medical University Hospital, China Medical University, Taichung 40402, Taiwan

Corresponding author: Thabet Abdeljawad (tabdeljawad@psu.edu.sa)

ABSTRACT Root-finding of non-linear equations is one of the most appearing problems in engineering sciences. Most of the complicated engineering problems can be modeled easily by means of non-linear functions. The role of iterative algorithms via computers for solving such functions is much important and cannot be denied in the modern age. In an iterative algorithm, the convergence order and the computational cost per iteration are the main characteristics that depict its efficiency and performance i.e., a method with higher-order and lower computational cost will be more efficient and vice versa. Keeping these facts into consideration, the main goal of this paper is to introduce a new derivative-free iterative method that performs better. We develop this algorithm by utilizing the forward- and finite-difference schemes on well-known Householder's method, resulting in an efficient and derivative-free algorithm with a low per iteration computing cost. We also look at the developed algorithm's convergence criterion and show that it is quartic-order convergent. We investigate nine test-examples and solve them to demonstrate its correctness, validity, and efficiency numerically. Some real-world engineering problems in civil and chemical engineering are also included in these examples. The numerical results of the test-examples reveal that the newly constructed method outperforms the existing similar algorithms found in the literature. We consider various different-degrees complex polynomials for the graphical analysis and used a computer tool to create the polynomiographs of the proposed quartic-order algorithm and compare it to other comparable existing approaches. The graphical findings show that the developed method has a faster convergence speed than the other comparable algorithms.

INDEX TERMS Order of convergence, non-linear equations, Traub's method, Householder's method, polynomiography.

I. INTRODUCTION

In today's world, the importance of computers in applied mathematics cannot be overstated. A variety of complicated problems may be readily addressed using various computer programmes such as Mathematica, Matlab, Maple, and others. Mathematicians have been using computers excessively in several branches of mathematics in recent years, especially in polynomials' root-finding, which has played a key role in different modern disciplines. In engineering, root-finding algorithms can be used to solve a variety of complicated problems after converting them into the form of non-linear scalar equations. We need iterative algorithms to solve these types of engineering problems, since analytical approaches do not always work. Newton [1] presented the well-known classical iterative method in the late fifteenth

century. Researchers enhanced the order and efficiency of existing methods in the contemporary era, and multi-step algorithms were established. For further information, see [2]–[11]. Multi-step methods usually have a larger computing cost due to the higher derivatives involved, which is the main disadvantage of these algorithms. It is difficult to maintain a balance between computing cost and convergence order since increasing one reduces the other.

Researchers have improved the current iteration schemes by applying various mathematical methodologies in recent years and have proposed some novel multi-step methods. In [12], Traub suggested the following quartic-order two-step algorithm:

$$\begin{aligned} v_p &= u_p - \frac{\psi(u_p)}{\psi'(u_p)}, \quad p = 0, 1, 2, \dots, \\ u_{p+1} &= u_p - \frac{\psi(u_p)}{\psi'(u_p)}, \end{aligned} \quad (1)$$

The associate editor coordinating the review of this manuscript and approving it for publication was Santi Pavone.

which is usually known as Traub’s method (TM), where ψ is the real-valued function defined on the open interval of \mathbb{R} .

Noor et al. [13] in 2012, developed the following iteration formula:

$$v_p = u_p - \frac{\psi(u_p)}{\psi'(u_p)}, \quad p = 0, 1, 2, \dots,$$

$$u_{p+1} = u_p - \frac{\psi(u_p)}{\psi'(u_p)} + \left(\frac{\psi(u_p)}{\psi'(u_p)} \right) \frac{\psi'(v_p)}{\psi'(u_p)}, \quad (2)$$

which is called Noor’s method (NRM) for computing the approximate solution of non-linear equations.

Zhanlav et al. [14] proposed a three-step predictor-corrector iterative approach in 2010, which is as follows:

$$v_p = u_p - \frac{\psi(u_p)}{\psi'(u_p)}, \quad p = 0, 1, 2, \dots,$$

$$w_p = v_p - \frac{\psi(v_p)}{\psi'(u_p)},$$

$$u_{p+1} = v_p - \frac{\psi(v_p) + \psi(w_p)}{\psi'(u_p)}. \quad (3)$$

The above iterative scheme is a quartic-order root-finding algorithm and is known as Zhanlav’s method (ZM).

In this paper, we provide a novel 4th order and derivative-free technique for addressing engineering problems. The forward- and finite-difference techniques on the Householder’s approach were used to build this algorithm. We further verify that the developed method has 4th order convergence. By applying it to a variety of real-world engineering problems, we show its superiority upon other similar existing algorithms in the literature. Using computer technology, a dynamical comparison of the created approach with other related algorithms was also given.

The remaining parts of the paper are divided as follows. In Section 2, an efficient and derivative-free method has been developed. The developed algorithm’s convergence criterion was addressed in Section 3. Nine random and technical problems were solved in Section 4. In Section 5, a graphical characterization of the suggested method is shown. Finally, Section 6 contains the paper’s conclusion.

II. CONSTRUCTION OF A NEW ALGORITHM

Let $\psi : D \rightarrow \mathbb{R}$, $D \subset \mathbb{R}$ be a one-variable function, then by using Taylor’s series expansion, Householder [15] proposed the following iterative scheme:

$$u_{p+1} = u_p - \frac{\psi(u_p)}{\psi'(u_p)} - \frac{\psi^2(u_p)\psi''(u_p)}{2\psi'^3(u_p)}, \quad p = 0, 1, 2, \dots, \quad (4)$$

which is a cubic-order Householder’s method. In [16], the authors presented the following modified form of Householder’s method:

$$v_p = u_p - \frac{\psi(u_p)}{\psi'(u_p)}, \quad p = 0, 1, 2, \dots,$$

$$u_{p+1} = v_p - \frac{\psi(v_p)}{\psi'(v_p)} - \frac{\psi^2(v_p)\psi''(v_p)}{2\psi'^3(v_p)}. \quad (5)$$

This is a two-step iteration method for computing scalar non-linear equations’ zeros. The biggest downside of the aforementioned method is its high computational cost per iteration since it uses first and second derivatives to execute. It should also be noted that the first derivative in the above iteration scheme exists in the denominator and hence it cannot be applied to those functions whose first derivative becomes zero at some specific points in domain. We estimate the first and second derivatives of the aforementioned method and make it derivative-free to reduce its computing cost and make it more effective and adaptable.

To estimate second derivative $\psi''(v)$, we utilize the finite-difference approximation as:

$$\psi''(v_p) = \frac{\psi'(v_p) - \psi'(u_p)}{\psi(v_p) - \psi(u_p)}. \quad (6)$$

In order to estimate first derivatives, we utilized forward-difference scheme [17], [18] as follows:

$$\psi'(u_p) = \frac{\psi(u_p + \psi(u_p)) - \psi(u_p)}{\psi(u_p)} = \eta(u_p). \quad (7)$$

$$\psi'(v_p) = \frac{\psi(v_p + \psi(v_p)) - \psi(v_p)}{\psi(v_p)} = \zeta(v_p). \quad (8)$$

Using (7) and (8) in (6)

$$\psi''(v_p) = \frac{\zeta(v_p) - \eta(u_p)}{\psi(v_p) - \psi(u_p)} = \rho(u_p, v_p). \quad (9)$$

With the help of equations. (7)–(9) in (5), we achieve a novel two-step algorithm as given below:

Algorithm 1: For the initial guess u_0 , find the approximate solution u_{p+1} using the iteration schemes as given below:

$$v_p = u_p - \frac{\psi(u_p)}{\eta(u_p)}, \quad p = 0, 1, 2, \dots,$$

$$u_{p+1} = v_p - \frac{\psi(v_p)}{\zeta(v_p)} - \frac{\psi^2(v_p)\rho(u_p, v_p)}{2\zeta^3(v_p)}.$$

This is a two-step novel iterative approach for obtaining approximate roots (zeros) of non-linear functions in one variable that is derivative-free. The basic and important feature of the aforementioned algorithm is its large applicability area because it does not need any derivative of the function and thus covers those functions for which other methods have failed to find the approximated root because most methods have these derivatives in the denominator and these methods do not work when they become zero at certain points in the domain. The substitution of the first and second derivatives in (5) lowers the computational cost per iteration, resulting in a new derivative-free algorithm with a higher efficiency index than conventional iteration schemes. The numerical results of test-examples reveal that the proposed technique outperforms the other comparable algorithms in the literature.

III. CONVERGENCE ANALYSIS

This section contains the convergence analysis of the designed method i.e., Algorithm 1.

Theorem 1: Suppose β be the simple root of the Eq. $\psi(u) = 0$, where $\psi(u)$ is differentiable near the actual root β , then the convergence order of Algorithm 1 is four. Furthermore, it fulfills the following error equation:

$$e_{p+1} = \mathbb{A}e_p^4 + O(e^5),$$

where $\mathbb{A} = 2(\frac{\psi''(\beta)}{2\psi'(\beta)})^3$.

Proof: To show the convergence of Algorithm 1, we suppose that e_p is the p th iteration's error, where $e_p = u_p - \beta$ and by the Taylor series about $u = \beta$, we have:

$$\begin{aligned} \psi(u_p) &= \psi'(\beta)e_p + \frac{1}{2!}\psi''(\beta)e_p^2 + \frac{1}{3!}\psi'''(\beta)e_p^3 \\ &\quad + \frac{1}{4!}\psi^{(iv)}(\beta)e_p^4 + O(e_p^5), \\ \psi(u_p) &= \psi'(\beta)[e_p + b_2e_p^2 + b_3e_p^3 + b_4e_p^4 + O(e_p^5)], \quad (10) \\ \eta(u_p) &= \psi'(\beta)[1 + 3b_2e_p + (7b_3 + b_2^2)e_p^2 + (6b_2b_3 \\ &\quad + 15b_4)e_p^3 + (18b_2b_4 + 31b_5 + b_3b_2^2 + 5b_3^2)e_p^4 \\ &\quad + O(e_p^5)], \quad (11) \end{aligned}$$

where

$$b_p = \frac{1}{p!} \frac{\psi^{(p)}(\beta)}{\psi'(\beta)}.$$

Using (10) and (11), we obtain:

$$v_p = \psi'(\beta)[\beta + 2b_2e_p^2 + (6b_3 - 5b_2^2)e_p^3 + (-26b_2b_3 + 13b_2^3 + 14b_4)e_p^4 + O(e_p^5)], \quad (12)$$

$$\psi(v_p) = \psi'(\beta)[2b_2e_p^2 + (6b_3 - 5b_2^2)e_p^3 + (17b_2^3 - 26b_2b_3 + 14b_4)e_p^4 + O(e_p^5)], \quad (13)$$

$$\zeta(v_p) = \psi'(\beta)[1 + 6b_2^2e_p^2 + (18b_2b_3 - 15b_2^3)e_p^3 + (-50b_3b_2^2 + 43b_2^4 + 42b_2b_4)e_p^4 + O(e_p^5)], \quad (14)$$

$$\begin{aligned} \rho(u_p, v_p) &= \psi'(\beta)[3b_2 + (7b_3 - 2b_2^2)e_p + (10b_2b_3 + 15b_4 \\ &\quad - 2b_2^3)e_p^2 + (-62b_3b_2^2 + 16b_2^4 + 30b_2b_4 + 31b_5 \\ &\quad + 40b_2^2)e_p^3 + (78b_3b_2 - 258b_2b_3^2 + 188b_3b_4 \\ &\quad + 237b_3b_2^2 - 64b_2^2b_4 + 63b_6 - 48b_2^5)e_p^4 \\ &\quad + O(e_p^5)]. \quad (15) \end{aligned}$$

Using equations (10)–(15) in Algorithm 1, we have:

$$u_{p+1} = \beta + 2b_2^3e_p^4 + O(e_p^5),$$

which implies that

$$e_{p+1} = 2b_2^3e_p^4 + O(e_p^5). \quad (16)$$

The above equation confirmed the quartic-order convergence Algorithm 1. \square

IV. NUMERICAL COMPARISONS AND REAL-WORLD APPLICATIONS

In this part of the paper, we provide five random and four real-world problems in engineering to illustrate the correctness, validity, and robust performance of the newly constructed iteration method. By analyzing several non-linear

problems given below, we compare the proposed method with Noor's method (NRM) [13], Traub's method (TM) [12] and Zhanlav's method (ZM) [14].

Example 1: Blood Rheology Model

Blood rheology is a discipline of science that studies the physical and flow characteristics of blood [19]. Blood is a non-Newtonian fluid that is classified as Caisson. To investigate the plug flow of Caisson fluids, we used the following non-linear function:

$$H = 1 - \frac{16}{7}\sqrt{u} + \frac{4}{3}u - \frac{1}{21}u^4 \quad (17)$$

where H calculates the flow-rate reduction. Using $H = 0.40$ in (17), we get the following expression:

$$\psi_1(u) = \frac{1}{441}u^8 - \frac{8}{63}u^5 - 0.05714285714u^4 + \frac{16}{9}u^2 - 3.624489796u + 0.3. \quad (18)$$

The initial guess for starting the iteration process to solve $\psi_1(u) = 0$ was $u_0 = 0.9$, and the corresponding results are shown in Table 1.

Example 2: To Calculate Volume from van der Waal's Equation

The renowned van der Waal's equation, established by van der Waal [20], is used in engineering to investigate the gas behavior, as seen below:

$$(P + \frac{K_1n^2}{V^2})(V - nK_2) = nRT. \quad (19)$$

We get the following non-linear equation by taking feasible values for the appearing parameters in (19):

$$\psi_2(u) = 0.986u^3 - 5.181u^2 + 9.067u - 5.289,$$

where u signifies volume and may be readily calculated by solving $\psi_2(u) = 0$. Because ψ_2 is a cubic-degree polynomial, three roots (zeros) must exist, and one of them is 1.9298462428, that is only feasible because the volume of the gas is always positive. To begin the iteration procedure, we take $u_0 = 0.10$ and the results are inserted into Table 1.

Example 3: Open Channel Flow Problem

In fluid dynamics, Manning's equation [21] describes water-flow in an open channel with uniform flow having the following mathematical expression:

$$\text{Flow of Water} = \Phi = \frac{\sqrt{SAR}^{\frac{2}{3}}}{N}. \quad (20)$$

In (20), R , S , A , and N symbolize hydraulic-radius, slope, area, and Manning's roughness coefficient respectively. For a rectangular shaped channel with depth u and breadth b , we have the following expression:

$$A = bu, \quad \text{and} \quad R = \frac{bu}{b + 2u}.$$

We get the following results by inserting these values into (20):

$$\Phi = \frac{\sqrt{S}bu}{N} \left(\frac{bu}{b + 2u} \right)^{\frac{2}{3}}.$$

Table 1 presents a comprehensive comparison of some well-known existing iterative methods with the proposed iteration technique. The table’s columns include information such as the number of used iterations, the final estimated root, the absolute functional value at that root, and the positive distance of the two successive estimates.

The acquired test-examples’ results, which are presented in Table 1, demonstrate the effectiveness of the proposed method with respect to the other similar existing methods. In the last example, we took such a non-linear function in which its first derivative has become zero at the initial guess, and the results showed that all the comparable methods did not work in this condition but the suggested method still worked and determined the approximated root up to the given accuracy, which is the plus point of the proposed method, and justifies its better applicability among the other methods in comparison. Based on the overall numerical findings, we can assert that the devised algorithm has a robust performance in terms of accuracy, speed, number of iterations, and computational cost and that it is preferable to the other comparable root-finding methods.

V. POLYNOMIOGRAPHY

Kalantari [23], [24] coined the term polynomiography in 2005, defining it as a way of producing aesthetically beautiful graphical things by utilizing the mathematical convergence properties of iteration functions like Newton’s iterative algorithm [25], Halley’s iterative algorithm [26] and Householder’s iterative algorithm [15], etc. Polynomiographs are pictorial objects created as a result of polynomiography.

In order to draw the polynomiographs on the complex plane \mathbb{C} using a computer programme, we pick a rectangle $R \in \mathbb{C}$ of dimension $[-2, 2] \times [-2, 2]$, with the accuracy $\varepsilon = 0.001$ and maximal iterations $L = 20$. This rectangle comprises the roots (zeros) of a polynomial and corresponds to the beginning point z_0 in rectangle R , we commence the iteration-procedure and give a color to the point corresponding to z_0 . The color black is allocated to the spots where the algorithm diverges. The quality of created graphical objects is determined by the discretization of rectangle R . For example, if we discretize R into a 2000×2000 grid, the displayed polynomiographs will have the higher resolution and picture quality.

According to the theorem of Algebra, if q is an n th-degree polynomial, it must have the n th number of zeros and may be represented as:

$$q(z) = d_n z^n + d_{n-1} z^{n-1} + \dots + d_1 z + d_0. \tag{22}$$

If $z_1, z_2, \dots, z_{n-1}, z_n$ are the zeros (roots) of the complex polynomial q , then (22) may be expressed as:

$$q(z) = (z - z_1)(z - z_2) \dots (z - z_p), \tag{23}$$

where $\{d_n, d_{n-1}, \dots, d_1, d_0\}$ are the complex coefficients.

For drawing graphical objects, any method involving iteration can be used to the aforementioned representations of q . The basic algorithm for drawing polynomiographs is presented in Algorithm 2.

Algorithm 2 Plotting of Polynomiograph

Input: $q \in \mathbb{C}[Z]$ — Complex Polynomial, $\mathfrak{A} \subset \mathbb{C}$ — Area, L — Upper Bound of Iterations, τ — Iteration Formula, ε — Accuracy, Colormap $[0 \dots C - 1]$ — Colormap With C Colors.

Output: Polynomiograph for the given complex-polynomial.

```

for  $z_0 \in \mathfrak{A}$  do
     $p = 0$ 
    while  $p \leq L$  do
         $z_{p+1} = \tau(z_p)$ 
        if  $|z_{p+1} - z_p| < \varepsilon$  then
            break
         $p = p + 1$ 
    color  $z_0$  via colormap.
    
```

In the aforementioned algorithm, i.e., Algorithm 2, an iteration procedure is considered to be converged if the Convergence Test $(z_{p+1}, z_p, \varepsilon)$ returns TRUE and vice versa. The usual test for identifying an algorithm’s convergence or divergence is provided as:

$$|z_{p+1} - z_p| < \varepsilon. \tag{24}$$

The symbol $\varepsilon > 0$ in the above relation denotes accuracy, whereas z_p, z_{p+1} are the two successive estimations in the process of iteration. We also used (24) as a stopping criterion in this paper. By adjusting the parameters ε, L , and the iteration technique, a number of various colored graphical objects may be plotted. For more information on polynomiography and its implementations, one may concern [27]–[39] and the references are cited therein.

For drawing graphical objects in the complex plane, we use the four complex polynomials listed below:

$$q_1(z) = z^2 - 1, \quad q_2(z) = z^2 - z + 1, \\ q_3(z) = z^3 - 1, \quad q_4(z) = z^3 - z^2 + z - 1.$$

We use the colormap shown in Figure 1 to color the iterations.



FIGURE 1. The colormap for coloring iterations.

Example 6: Polynomiographs for q_1 Using Different Iteration Methods

In the 6th example, we analyze and compare the dynamical results achieved by various iteration methods with the newly devised quartic-order method for the quadratic-degree polynomial $z^2 - 1$ having two distinct roots: 1 and -1 . To obtain the simple roots, we ran all root-finding methods, and the results are shown in Figure 2.

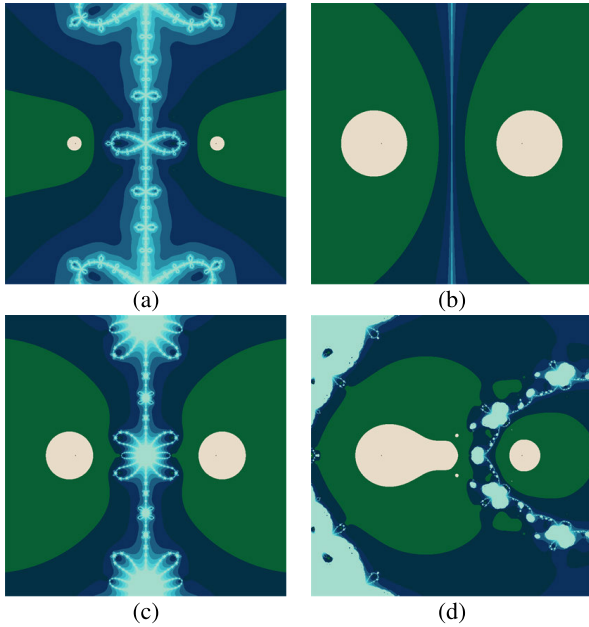


FIGURE 2. Polynomiographs corresponding to the 2nd-degree polynomial q_1 . (a) Shows NRM, (b) shows TM, (c) shows ZM, and (d) shows Algorithm 1.

Example 7: Polynomiographs for q_2 Using Different Iteration Methods

In the 7th example, we analyze and compare the dynamical results achieved by various iteration techniques with the newly devised quartic-order algorithm for the quadratic-degree polynomial $z^2 - z + 1$ which possesses two distinct simple zeros: $\frac{1}{2} + \frac{1}{2}\sqrt{3}i$ and $\frac{1}{2} - \frac{1}{2}\sqrt{3}i$. We executed all root-finding algorithms to achieve the simple roots and the results can be visualized in Figure 3.

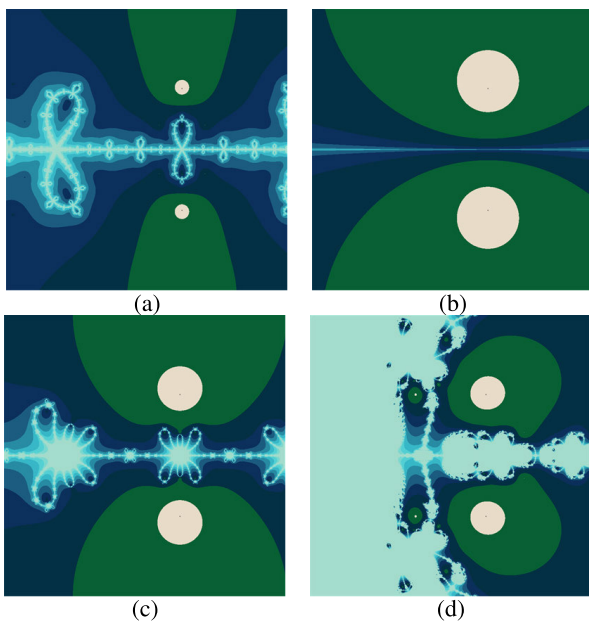


FIGURE 3. Polynomiographs corresponding to the 2nd-degree polynomial q_2 . (a) Shows NRM, (b) shows TM, (c) shows ZM, and (d) shows Algorithm 1.

Example 8: Polynomiographs for q_3 Using Different Iteration Methods

In the 8th example, we analyze and compare the dynamical results achieved by various iteration methods with the newly devised quartic-order method for the cubic-degree polynomial $z^3 - 1$, which has three unique zeros: 1 , $-\frac{1}{2} - \frac{\sqrt{3}}{2}i$, $-\frac{1}{2} + \frac{\sqrt{3}}{2}i$. We used all root-finding methods to determine simple roots, and the results are shown in Figure 4.

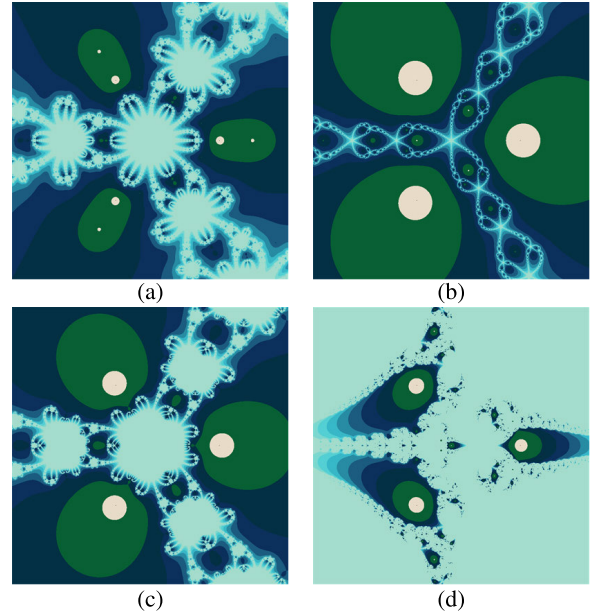


FIGURE 4. Polynomiographs corresponding to 3rd-degree polynomial q_3 . (a) Shows NRM, (b) shows TM, (c) shows ZM, and (d) shows Algorithm 1.

Example 9: Polynomiographs for q_4 Using Different Iteration Methods

In the 9th example, we take polynomial $z^3 - z^2 + z - 1$, which has unique roots 1 , i , and $-i$. We created the visually attractive graphical objects by running all the comparable root-finding methods and the results are shown in Figure 5.

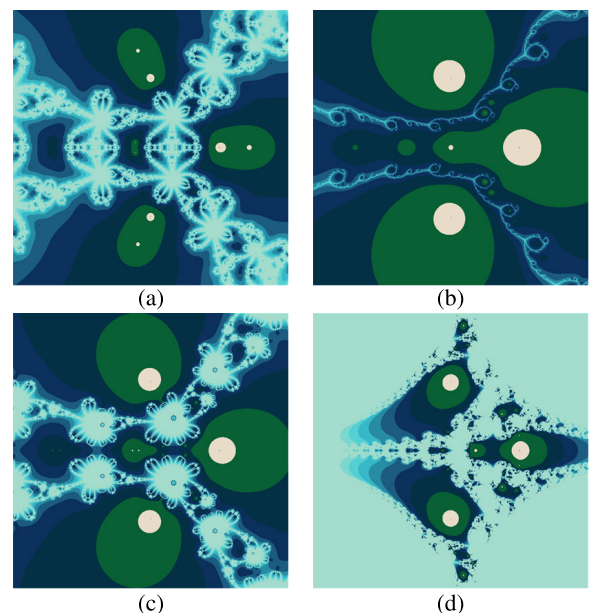


FIGURE 5. Polynomiographs corresponding to the 3rd-degree polynomial q_4 . (a) Shows NRM, (b) shows TM, (c) shows ZM, and (d) shows Algorithm 1.

In Examples 6-9, we used Mathematica 12.0 to run all of the comparable root-finding methods for constructing aesthetically beautiful polynomiographs. We can study easily the graphical aspects and stability of different iteration methods using the resulting graphical objects. One can easily observe that the newly constructed method has a significantly bigger convergence zone as compared to other comparable ones. The colors' shades demonstrate the algorithm's efficiency that has been used to draw the polynomiograph. These graphical objects indicate two important aspects: the convergence speed and the dynamics of the iterative methods used to build these graphics. The first may be shown by examining the image's color tones. Color richness in graphical objects demonstrates robust convergence with fewer iterations. The second attribute may be examined by observing the color variations of the drawn polynomiographs. The regions with minor color fluctuation represent low dynamic zones, whereas the zones of substantial color variation indicate high dynamic zones. The same color-zones signify the similar No. of iterations used by different methods to seek the approximated solution up to a particular precision, and their graphical representation of the contour lines on the map is equivalent.

VI. CONCLUDING REMARKS

Based on forward- and finite-difference schemes, we developed a novel root-finding algorithm for non-linear equations that is more efficient and derivative-free. We reviewed the proposed algorithm's convergence criterion and demonstrated its quartic-order convergence. We assumed certain random and engineering problems, turned them into non-linear functions, and then solved them to illustrate the applicability and robust performance of the described method. The numerical findings in Table 1 demonstrate that the newly proposed method outperforms the previous comparable root-finding algorithms in terms of convergence, time, accuracy, and computational order of convergence. The proposed algorithm's robustness can also be anticipated by looking at the accuracy of consecutive estimations, which is significantly higher than that of other comparable algorithms. To investigate the complicated dynamic nature of the proposed method, polynomiographs for several complex polynomials were created using the computer application Mathematica 12.0. The resulting graphical objects are unique and visually beautiful, revealing the devised algorithm's graphical properties and superior convergence rate over the other comparable algorithms. Using the same concept as in this article, a new family of derivative-free techniques for calculating the roots of non-linear equations may be constructed.

ACKNOWLEDGMENT

The author Thabet Abdeljawad would like to thank Prince Sultan University for the support through the TAS Research Laboratory.

REFERENCES

- [1] D. E. Kincaid and E. W. Cheney, *Numerical Analysis*. Pacific Grove, CA, USA: Brooks/Cole Publishing Company, 1990.
- [2] A. R. Alharbi, M. I. Faisal, F. A. Shah, M. Waseem, R. Ullah, and S. Sherbaz, "Higher order numerical approaches for nonlinear equations by decomposition technique," *IEEE Access*, vol. 7, pp. 44329–44337, 2019, doi: [10.1109/ACCESS.2019.2906470](https://doi.org/10.1109/ACCESS.2019.2906470).
- [3] A. Naseem, M. A. Rehman, T. Abdeljawad, and Y.-M. Chu, "Novel iteration schemes for computing zeros of non-linear equations with engineering applications and their dynamics," *IEEE Access*, vol. 9, pp. 92246–92262, 2021, doi: [10.1109/ACCESS.2021.3091473](https://doi.org/10.1109/ACCESS.2021.3091473).
- [4] F. Soleimani, F. Soleymani, and S. Shateyi, "Some iterative methods free from derivatives and their basins of attraction for nonlinear equations," *Discrete Dyn. Nature Soc.*, vol. 2013, pp. 1–10, Apr. 2013, doi: [10.1155/2013/301718](https://doi.org/10.1155/2013/301718).
- [5] S. Abbasbandy, "Improving Newton–Raphson method for nonlinear equations by modified adomian decomposition method," *Appl. Math. Comput.*, vol. 145, nos. 2–3, pp. 887–893, Dec. 2003.
- [6] A. Naseem, M. A. Rehman, and T. Abdeljawad, "Higher-order root-finding algorithms and their basins of attraction," *J. Math.*, vol. 2020, pp. 1–11, Nov. 2020.
- [7] R. Behl and E. Martínez, "A new high-order and efficient family of iterative techniques for nonlinear models," *Complexity*, vol. 2020, pp. 1–11, Jan. 2020, doi: [10.1155/2020/1706841](https://doi.org/10.1155/2020/1706841).
- [8] M. S. Rhee, Y. I. Kim, and B. Neta, "An optimal eighthorder class of three-step weighted Newton's methods and their dynamics behind the purely imaginary extraneous fixed points," *Int. J. Comput. Math.*, vol. 95, p. 11, 2174–2211, 2018, doi: [10.1080/00207160.2017.1367387](https://doi.org/10.1080/00207160.2017.1367387).
- [9] A. Naseem, M. A. Rehman, T. Abdeljawad, and F. Balibrea, "Numerical algorithms for finding zeros of nonlinear equations and their dynamical aspects," *J. Math.*, vol. 2020, pp. 1–11, Sep. 2020.
- [10] D. Cebic, M. Paunovic, and N. M. Ralevic, "A variant of sharma-Arora's optimal eighth-order family of methods for finding a simple root of nonlinear equation," in *Proc. IEEE 16th Int. Symp. Intell. Syst. Informat. (SISY)*, Sep. 2018, pp. 000081–000086, doi: [10.1109/SISY.2018.8524857](https://doi.org/10.1109/SISY.2018.8524857).
- [11] A. Naseem, M. A. Rehman, T. Abdeljawad, and Y.-M. Chu, "Some engineering applications of newly constructed algorithms for one-dimensional non-linear equations and their fractal behavior," *J. King Saud Univ. Sci.*, vol. 33, no. 5, Jul. 2021, Art. no. 101457, doi: [10.1016/j.jksus.2021.101457](https://doi.org/10.1016/j.jksus.2021.101457).
- [12] J. F. Traub, *Iterative Methods for the Solution of Equations*. New York, NY, USA: Chelsea Publishing Company, 1982.
- [13] M. A. Noor, W. A. Khan, and A. Hussain, "A new modified Halley method without second derivatives for nonlinear equation," *Appl. Math. Comput.*, vol. 189, no. 2, pp. 1268–1273, Jun. 2007.
- [14] T. Zhanlav, O. Chuluunbaatar, and G. Ankhbayar, "On Newton-type methods with fourth and fifth-order convergence," *Discrete Contin. Models Appl. Comput. Sci.*, vol. 2, no. 2, pp. 30–35, 2010.
- [15] A. S. Householder, *The Numerical Treatment of a Single Nonlinear Equation*. New York, NY, USA: McGraw-Hill, 1970.
- [16] W. Nazeer, M. Tanveer, S. M. Kang, and A. Naseem, "A new householder's method free from second derivatives for solving nonlinear equations and polynomiography," *J. Nonlinear Sci. Appl.*, vol. 9, pp. 998–1007, Jan. 2016.
- [17] P. Jain, "Steffensen type methods for solving non-linear equations," *Appl. Math. Comput.*, vol. 194, no. 2, pp. 527–533, Dec. 2007.
- [18] S. D. Conte and C. D. Boor, *Elementary Numerical Analysis: An Algorithmic Approach*. New York, NY, USA: McGraw-Hill, 1981.
- [19] R. L. Fournier, *Basic Transport Phenomena in Biomedical Engineering*. New York, NY, USA: Taylor & Francis, 2007, pp. 1–611.
- [20] V. D. Waals and J. Diderik, "Over de continuïteit van den gasen vloeïstoftoestand (on the continuity of the gas and liquid state)," M.S. thesis, Leiden Univ., Leiden, The Netherlands, 1873.
- [21] R. Manning, "On the flow of water in open channels and pipes," *Trans. Inst.*, vol. 20, pp. 161–207, 1891.
- [22] M. Planck, *The Theory of Heat Radiation*, M. Masius, Ed. PA, USA: P. Blakiston's Son, 1914.
- [23] B. Kalantari, "Method of creating graphical works based on polynomials," U.S. Patent 6894705, May 17, 2005.
- [24] B. Kalantari, "Polynomiography: From the fundamental theorem of algebra to art," *Leonardo*, vol. 38, no. 3, pp. 233–238, Jun. 2005.
- [25] R. L. Burden and J. D. Faires, *Numerical Analysis*. Pacific Grove, CA, USA: Brooks/Cole, 1997.

- [26] D. Chen, I. K. Argyros, and Q. S. Qian, "A note on the Halley method in Banach spaces," *Appl. Math. Comput.*, vol. 58, nos. 2–3, pp. 215–224, Oct. 1993.
- [27] B. Kalantari, "An invitation to polynomiography via exponential series," 2017, *arXiv:1707.09417*.
- [28] B. Kalantari and E. H. Lee, "Newton–Ellipsoid polynomiography," *J. Math. Arts*, vol. 13, no. 4, pp. 336–352, Nov. 2019.
- [29] I. Gościński and K. Gdawiec, "Visual analysis of dynamics behaviour of an iterative method depending on selected parameters and modifications," *Entropy*, vol. 22, no. 7, p. 734, Jul. 2020.
- [30] A. Naseem, M. A. Rehman, and T. Abdeljawad, "Computational methods for non-linear equations with some real-world applications and their graphical analysis," *Intell. Autom. Soft Comput.*, vol. 30, no. 3, pp. 805–819, 2021, doi: [10.32604/iasc.2021.019164](https://doi.org/10.32604/iasc.2021.019164).
- [31] W. Kotarski, K. Gdawiec, and A. Lisowska, "Polynomiography via Ishikawa and Mann iterations," *Advances in Visual Computing (Lecture Notes in Computer Science)*, vol. 7431, G. Bebis, R. Boyle, B. Parvin, Eds. Berlin, Germany: Springer, 2012, pp. 305–313.
- [32] K. Gdawiec, "Fractal patterns from the dynamics of combined polynomial root finding methods," *Nonlinear Dyn.*, vol. 90, no. 4, pp. 2457–2479, 2017.
- [33] K. Gdawiec, W. Kotarski, and A. Lisowska, "Visual analysis of the Newton's method with fractional order derivatives," *Symmetry*, vol. 11, no. 9, p. 1143, Sep. 2019.
- [34] A. Naseem, M. A. Rehman, and T. Abdeljawad, "Numerical methods with engineering applications and their visual analysis via polynomiography," *IEEE Access*, vol. 9, pp. 99287–99298, 2021, doi: [10.1109/ACCESS.2021.3095941](https://doi.org/10.1109/ACCESS.2021.3095941).
- [35] M. Scott, B. Neta, and C. Chun, "Basin attractors for various methods," *Appl. Math. Comput.*, vol. 218, no. 6, pp. 2584–2599, Nov. 2011.
- [36] J. R. Sharma and H. Arora, "A new family of optimal eighth order methods with dynamics for nonlinear equations," *Appl. Math. Comput.*, vol. 273, pp. 924–933, Jan. 2016.
- [37] A. Naseem, M. A. Rehman, and J. Younis, "A new root-finding algorithm for solving real-world problems and its complex dynamics via computer technology," *Complexity*, vol. 2021, pp. 1–10, Nov. 2021, doi: [10.1155/2021/6369466](https://doi.org/10.1155/2021/6369466).
- [38] Y. C. Kwun, M. Tanveer, W. Nazeer, K. Gdawiec, and S. M. Kang, "Mandelbrot and Julia sets via Jungck-CR iteration with s -convexity," *IEEE Access*, vol. 7, pp. 12167–12176, 2019, doi: [10.1109/ACCESS.2019.2892013](https://doi.org/10.1109/ACCESS.2019.2892013).
- [39] A. Naseem, M. A. Rehman, and T. Abdeljawad, "Some new iterative algorithms for solving one-dimensional non-linear equations and their graphical representation," *IEEE Access*, vol. 9, pp. 8615–8624, 2021, doi: [10.1109/ACCESS.2021.3049428](https://doi.org/10.1109/ACCESS.2021.3049428).



AMIR NASEEM received the Ph.D. degree in mathematics from the University of Management and Technology, Lahore, Pakistan. He has published 35 research articles in different well-known journals. His current research interests include numerical analysis, iterative methods, and polynomiography.



M. A. REHMAN received the Ph.D. degree in mathematics from Government College University, Lahore, Pakistan. He is currently a Full Professor with the University of Management and Technology, Lahore. He has published over 60 research articles in different international journals. His main research interests include modeling, graph theory, and mathematical analysis.



THABET ABDELJAWAD received the Ph.D. degree in mathematics from Middle East Technical University, in 2000. He is currently a Full Professor with Prince Sultan University. He has published more than 450 research articles in different well-known journals. He has been listed as a highly cited researcher for the three consecutive years 2019, 2020, and 2021. He has published more than 450 research articles in different well-known journals. His main research interests include fractional calculus, discrete fractional operators, metric spaces, and fixed point theory.

• • •