Research Article

Inverse Numerical Iterative Technique for Finding all Roots of Nonlinear Equations with Engineering Applications

Mudassir Shams,1 Naila Rafiq,2 Babar Ahmad,3 and Nazir Ahmad Mir1

1Department of Mathematics and Statistics, Riphah International University I-14, Islamabad 44000, Pakistan
2Department of Mathematics, NUML, Islamabad, Pakistan
3Department of Mathematics, Comsats University Islamabad, Islamabad 44000, Pakistan

Correspondence should be addressed to Mudassir Shams; shams6262@gmail.com and Naila Rafiq; nrafiq@numl.edu.pk

Received 24 October 2020; Revised 24 November 2020; Accepted 2 December 2020; Published 4 January 2021

Academic Editor: Ghulam Mustafa

Copyright © 2021 Mudassir Shams et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We introduce here a new two-step derivate-free inverse simultaneous iterative method for estimating all roots of nonlinear equation. It is proved that convergence order of the newly constructed method is four. Lower bound of the convergence order is determined using Mathematica and verified with theoretical local convergence order of the method introduced. Some nonlinear models which are taken from physical and engineering sciences as numerical test examples to demonstrate the performance and efficiency of the newly constructed modified inverse simultaneous methods as compared to classical methods existing in literature are presented. Dynamical planes and residual graphs are drawn using MATLAB to elaborate efficiency, robustness, and authentication in its domain.

1. Introduction

A wide range of problems in physical and engineering sciences can be formulated as a nonlinear equation:

\[ f(r) = 0. \]  (1)

The most ancient and popular iterative technique for approximating single roots of (1) is Newton’s method [1] which has local quadratic convergence:

\[ s^{(k)} = r^{(k)} - \frac{f(r^{(k)})}{f'(r^{(k)})}, \quad (k = 0, 1, \ldots). \]  (2)

Nedzibove et al., in [2], presented the inverse method of the same order corresponding to method (2):

\[ s^{(k)} = \frac{(r^{(k)})^2 f'(r^{(k)})}{r^{(k)} f'(r^{(k)}) + f(r^{(k)})}. \]  (3)

In the last few years, lot of work has been carried out on numerical iterative methods which approximate single root at a time of (1). There is another class of derivative-free iterative methods which approximates all roots of (1) simultaneously. The simultaneous iterative methods for approximating all roots of (1) are very popular due to their global convergence and parallel implementation on computer (see, e.g., Weierstrass [3], Kanno [4], Proinov [5], Petković [6], Mir [7], Nouriein [8], Aberth [9], and reference cited there in [10–22]).

Among derivative-free simultaneous methods, Weierstrass–Dochive [23] method (abbreviated as WDK) is the most attractive method given by

\[ s_i^{(k)} = r_i^{(k)} - \omega r_i^{(k)}, \]  (4)

where
Let the order of the IWM2 method is four. Nedzibove [2] introduced a new modification to (4), that is, an inverse method to WDK abbreviated as IWMK, i.e.,

\[ u_i^{(k)} = \frac{1}{r_i^{(k)}} \prod_{j 
eq i}^{n} (r_i^{(k)} - r_j^{(k)}) + f(r_i^{(k)}) \].

The main aim of this paper is to construct a two-step inverse method of convergence order four.

### 2. Construction of Family of Simultaneous Method for Distinct Roots

We modify the Weierstrass method (4) as follows:

\[ z_i^{(k+1)} = u_i^{(k)} - \frac{f(u_i^{(k)})}{\prod_{j 
eq i}^{n} (u_i^{(k)} - u_j^{(k)})} \],

where

\[ u_i^{(k)} = r_i^{(k)} - (f(r_i^{(k)})/\prod_{j 
eq i}^{n} (r_i^{(k)} - r_j^{(k)})) \]

and denote it by WDK2. Let us now convert method (7) into inverse iterative method as follows:

\[ z_i^{(k)} = \frac{1}{u_i^{(k)}} \prod_{j 
eq i}^{n} (u_i^{(k)} - u_j^{(k)}) + f(u_i^{(k)}) \]

where

\[ u_i^{(k)} = \left( \left( r_i^{(k)} \right)^2 \prod_{j 
eq i}^{n} (r_i^{(k)} - r_j^{(k)}) / \left( \prod_{j 
eq i}^{n} (r_i^{(k)} - r_j^{(k)}) + f(r_i^{(k)}) \right) \right) \]  

Thus, method (8) is a two-step inverse method abbreviated as IWM2.

### 2.1. Convergence Analysis

We prove here that convergence order of the IWM2 method is four.

Let \( D \subseteq C^n \) be an open convex subset, \( \Gamma: D \rightarrow C^n \) and \( m \) times differentiable operator \( (\Gamma_1(r), \ldots, \Gamma_n(r))^T \) be continuous, and the sequence \( (r^{(k)})_{k \in N} \) be defined by \( r^{(k+1)} = \Gamma(r^{(k)}) \);

\[ r^{(k)} = (r_1^{(k)}, \ldots, r_n^{(k)}) \]

\[ \Rightarrow r_i^{(k+1)} = \Gamma_i(r^{(k)}) \quad \forall i \in \{1, \ldots, n\}, k \in N, \]

where norm in \( C^n \) is defined as \( \|r\| = \max\{|r_1|, \ldots, |r_n|\} \).

**Theorem 1.** Let \( X \) and \( Y \) be normed spaces. Take an open convex subset \( D \) of \( X \) for a \( \alpha \) times Fréchet differential operator \( \Gamma \), i.e., \( \Gamma: D \rightarrow Y \). Then, for any \( x, y \in D \),

\[
\|\Gamma(y) - \Gamma(x) - \sum_{j=1}^{\alpha} f^{(j)}(x)\left( (y-x) \cdots (y-x) \right) \| \\
\leq \sup_{\xi \in (x,y)} \|r^{(\alpha)}(\xi)\|^q \\
\leq \frac{\|y-x\|^q}{q!} \quad \text{for } k \rightarrow 0.
\]
Thus, from inequality (14), \( (r)_{k\in N}^{(k)} \) is at least \( q \). Now, consider IWM2 as a vector function, i.e., \( \Gamma(r) = (\Gamma_1(r), \ldots, \Gamma_n(r)) \), where

\[
\Gamma_i(z_i) = \frac{(u_i)^2}{u_i + \left( f(u_i)/\prod_{j \neq i}^n (u_i - u_j) \right)}, \quad \text{where}
\]

\[
u_i = \frac{(r_i)^2}{r_i + \left( f(r_i)/\prod_{j \neq i}^n (r_i - r_j) \right)}.
\]

For a fixed point \( \beta = (\beta_1, \ldots, \beta_n) \), it is not difficult to prove \( (\partial \Gamma_i(\zeta)/\partial r_j) = (\partial^2 \Gamma_i(\zeta)/\partial r_i \partial r_j) = (\partial^3 \Gamma_i(\zeta)/\partial r_i^2 \partial r_j) = 0 \) and higher order partial derivative is not equal to zero. Thus, IWM2 has at least fourth-order convergence. \( \square \)

**Theorem 3.** Let \( \zeta_1, \ldots, \zeta_n \) be simple roots of (1) and for sufficiently close initial distinct estimations \( r^{(0)}_1, \ldots, r^{(0)}_n \) of the roots, respectively, IWM2 has then convergence order 4.

Proof. Consider \( \epsilon_i = r^{(k)}_i - \zeta_i \), \( \epsilon_i' = u_i^{(k)} - \zeta_i \), and \( \epsilon_i'' = z_i^{(k)} - \zeta_i \) be the errors in \( r^{(k)}_i \), \( u_i^{(k)} \), and \( z_i^{(k)} \), respectively. For simplicity, we omit iteration index \( k \). From first step of IWM2, we have

\[
u_i - \zeta_i = r_i - \zeta_i - \frac{\left( r_i f(r_i)/\prod_{j \neq i}^n (r_i - r_j) \right)}{r_i + \left( f(r_i)/\prod_{j \neq i}^n (r_i - r_j) \right)}.
\]

Thus, we obtain

\[
\epsilon_i' = \epsilon_i \left[ 1 - \left( \prod_{j \neq i}^n (r_i - \zeta_j)/(r_i - r_j) \right) \right] \left[ 1 + \left( f(r_i)/\prod_{j \neq i}^n (r_i - r_j) \right) \right]^{-1} = \epsilon_i \left[ \prod_{j \neq i}^n (r_i - \zeta_j)/(r_i - r_j) \right]^{-1} = \epsilon_i \left[ \prod_{j \neq i}^n (r_i - \zeta_j)/(r_i - r_j) \right]^{-1}.
\]

Using the expression \( \left( \prod_{j \neq i}^n (r_i - \zeta_j)/(r_i - r_j) \right) = 1 \), we have

\[
epsilon_i' = \epsilon_i \left[ \prod_{j \neq i}^n (r_i - \zeta_j)/(r_i - r_j) \right]^{-1} = \epsilon_i \left[ \prod_{j \neq i}^n (r_i - \zeta_j)/(r_i - r_j) \right]^{-1}.
\]

If we assume all errors are of the same order, i.e., \( |\epsilon_i| = |\epsilon_k| = O(|\epsilon|) \); then, we have

\[
epsilon_i'' = |\epsilon|^2 \left[ \frac{\left( 1/r_i \right) \prod_{j \neq i}^n (r_i - \zeta_j)/(r_i - r_j) - \sum_{k+1}^n (1/r_i - r_k) \prod_{j \neq i}^n (r_i - \zeta_k)/(r_i - r_j)}{1 + \left( \epsilon_i/r_i \right) \prod_{j \neq i}^n (r_i - \zeta_j)/(r_i - r_j)} \right] = O(|\epsilon|^2).
\]
From second-step of IWM2, we have

\[ z_i - \zeta_i = u_i - \zeta_i - \frac{u_i f(u_i) / \prod_{j \neq i}^{n} (u_i - u_j)}{r_i + f(u_i) / \prod_{j \neq i}^{n} (u_i - u_j)} . \]  
\[ \text{(21)} \]

Thus, we obtain

\[ \varepsilon_i'' = \varepsilon_i' \left[ 1 - \frac{\prod_{j \neq i}^{n} ((u_i - \zeta_j)/(u_i - u_j))}{1 + f(u_i) / \prod_{j \neq i}^{n} (u_i - u_j)} \right] \]
\[ = \varepsilon_i' \left[ 1 - \frac{\prod_{j \neq i}^{n} ((u_i - \zeta_j)/(u_i - u_j)) + (f(u_i) / \prod_{j \neq i}^{n} (u_i - u_j))}{1 + f(u_i) / \prod_{j \neq i}^{n} (u_i - u_j)} \right] . \]  
\[ \text{(22)} \]

As from the above argument \( \prod_{j \neq i}^{n} ((u_i - \zeta_j)/(u_i - u_j)) = 1 - \sum_{k \neq i}^{n} (\zeta_k/u_i - u_k) \prod_{j \neq i}^{k-1} ((u_i - \zeta_k)/(u_i - u_j)) \)

using in (22), we have

\[ \varepsilon_i'' = \varepsilon_i' \left[ \frac{(\varepsilon_i'/u_i) \left( \prod_{j \neq i}^{n} ((u_i - \zeta_j)/(u_i - u_j)) \right) - \sum_{k \neq i}^{n} (\zeta_k/u_i - u_k) \prod_{j \neq i}^{k-1} ((u_i - \zeta_k)/(u_i - u_j))}{1 + (\varepsilon_i'/u_i) \left( \prod_{j \neq i}^{n} ((u_i - \zeta_j)/(u_i - u_j)) \right)} \right] . \]  
\[ \text{(23)} \]

If we assume all errors are of the same order, i.e., \( |\varepsilon_i'| = |\varepsilon_i''| = O(|\varepsilon'|) \); then,

\[ |\varepsilon_i''| = |\varepsilon'|^2 \left[ \frac{(1/u_i) \prod_{j \neq i}^{n} ((u_i - \zeta_j)/(u_i - u_j)) - \sum_{k \neq i}^{n} (1/u_i - s_k) \prod_{j \neq i}^{k-1} ((u_i - \zeta_k)/(u_i - u_j))}{1 + (\varepsilon_i'/u_i) \prod_{j \neq i}^{n} ((u_i - \zeta_j)/(u_i - u_j))} \right] = O(|\varepsilon'|^2) \]
\[ = O\left( (|\varepsilon'|^2)^2 \right) = O(|\varepsilon'|^4). \]  
\[ \text{(24)} \]

Hence, the theorem is proved.

\[ \Box \]

2.1.1. Using CAS for Verification of Convergence Order. Consider

\[ f(r) = (r - \theta)(r - \phi)(r - \psi), \]  
\[ \text{(25)} \]

and the first component of \( \Gamma_i(r) \) iterative schemes to find zeros of (25), \( r^{(k+1)} = \Gamma(r^{(k)}) \), simultaneously. In order to verify Theorem 2 conditions, we have to express the differential of an operator \( \Gamma_i(r) \) in terms of their partial derivate of its component as \( \Gamma_i(r) \):
Modified Inverse Weierstrass Method:
\[
\Gamma_1(r_1, r_2, r_3) = \frac{(r)^3 \prod_{j \neq i} (r_i - r_j)}{r \prod_{j \neq i} (r_i - r_j) + f(r)},
\]

In [1]: \( D[\Gamma_1[r_1, r_2, r_3], r_1]/[r_1 \to \theta, r_2 \to \phi, r_3 \to \varphi] \),

Out [1]: 0,

In [2]: \( D[\Gamma_1[r_1, r_2, r_3], r_2]/[r_1 \to \theta, r_2 \to \phi, r_3 \to \varphi] \),

Out [2]: 0,

In [3]: Simplify \( D[\Gamma_1[r_1, r_2, r_3], r_3]/[r_1 \to \theta, r_2 \to \phi, r_3 \to \varphi] \),

Out [3]: \( \frac{1}{\theta + \phi} \)

and so on.

The lower bound of the convergence obtained until the first nonzero element of the row is found. The Mathematica code is given for each of the consider methods as follows.

Weierstrass–Dochive Method (WDK):
\[
\Gamma_1(r_1, r_2, r_3) := r - \frac{f(r)}{\prod_{j=1}^{n} (r_i - r_j)}, \quad (i, j = 1, \ldots, n),
\]

In [1] := \( D[\Gamma_1[r_1, r_2, r_3], r_1]/[r_1 \to \theta, r_2 \to \phi, r_3 \to \varphi] \),

Out [1] := 0,

In [2] := \( D[\Gamma_1[r_1, r_2, r_3], r_2]/[r_1 \to \theta, r_2 \to \phi, r_3 \to \varphi] \),

Out [2] := 0,

In [3] := Simplify \( D[\Gamma_1[r_1, r_2, r_3], r_3]/[r_1 \to \theta, r_2 \to \phi, r_3 \to \varphi] \),

Out [3] := \( \frac{-12}{\theta^3} \)

WDK2 Method:
\[
\Gamma_1(r_1, r_2, r_3) := u - \frac{f(u)}{\prod_{j=1}^{n} (u_i - u_j)},
\]

where \( u = r - (f(r))/\prod_{j \neq i} (r_i - r_j) \),

In [1] := \( D[\Gamma_1[r_1, r_2, r_3], r_1]/[r_1 \to \theta, r_2 \to \phi, r_3 \to \varphi] \),

Out [1] := 0,

In [13] := Simplify \( D[\Gamma_1[r_1, r_2, r_3], r_3]/[r_1 \to \theta, r_2 \to \phi, r_3 \to \varphi] \),

Out [13] := \( \frac{-12}{\theta^3} \)

IWM2 Method:
\[ \Gamma_1(r_1, r_2, r_3) = \frac{(u)^2 \prod_{j \neq i}^{n} (u_i - u_j)}{\prod_{j \neq i}^{n} (u_i - u_j) + f(u)} \]  \quad (31)

where \( u = (r) \prod_{j \neq i}^{n} (r_i - r_j)/(r) \prod_{j \neq i}^{n} (r_i - r_j) + f(r) \),

\[ \text{In [1]} = \frac{D[\Gamma_1[r_1, r_2, r_3], r_1]}{r_1 \longrightarrow \theta, r_2 \longrightarrow \phi, r_3 \longrightarrow \varphi} \]

Out [1] = 0,

\[ \therefore \]

\[ \text{In [14]} = \text{Simplify} \left[ \frac{D[\Gamma_1[r_1, r_2, r_3], r_1, r_1, r_1, r_1, r_1]}{r_1 \longrightarrow \theta, r_2 \longrightarrow \phi, r_3 \longrightarrow \varphi} \right] \]

Out [14] = \( \frac{24}{\theta^3} \)  \quad (32)

(1) Basins of Attraction. To provoke the basins of attraction of iterative schemes WDK, IWDK, WDK2, and IWM2 for the root of nonlinear equation, we execute the real and imaginary parts of the starting approximation as two axes over a mesh of 250 x 250 in complex plane. Using \( |r^{(k+1)} - r^{(k)}| < 10^{-3} \) as a stopping criteria and maximum number of iterations as 25. We allow different colors to mark to which root the iterative scheme converges and black in other case. Color brightness in basins shows less number of iterations. For the generation of basins, we consider the following four nonlinear functions, i.e., \( f_1(r) = \log r + e^r + 1 \) and \( f_2(r) = \sin((r - 1)/2) \cos((r - 3)/2) + 1 \).

The elapsed time from Table 1 and brightness in color in Figure 1(d)–2(d) show the dominance behavior of IWM2 over WDK, IWDK, and WDK2, respectively.

The elapsed time from Table 1 and brightness in color in Figure 2(d) show the dominance behavior of IWM2 over WDK, IWDK, and WDK2, respectively.

3. Numerical Results

Some nonlinear models from engineering and physical sciences are considered to illustrate the performance and efficiency of WDK2 and IWM2 using CAS Maple 18 with 64 digits floating point arithmetic for all computer calculations. We approximate the roots of (1) rather than the exact roots which depend on computer precision \( \epsilon \), and the following stopping criteria are used to terminate the computer program:

\[ e_1 = \| r^{(k+1)} - r^{(k)} \| < \epsilon, \]  \quad (33)

| Method | WDK | IWDK | WDK2 | IWM2 |
|--------|-----|------|------|------|
| \( f_1(r) \) | 0.12937 | 0.142207 | 0.323190 | 0.107267 |
| \( f_2(r) \) | 0.160921 | 0.23889 | 0.431936 | 0.153851 |

where \( e_i \) represents the absolute error. We take \( \epsilon = 10^{-30} \). In Tables 2–5, CO represents convergence order of iterative schemes WDK2 and IWM2, respectively.

3.1. Applications in Engineering. In this section, we discuss some applications in engineering.

Example 1 (see [24]). Fractional Conversion.

As expression described in [25, 26],

\[ f_3(r) = r^4 - 7.79075 r^3 + 14.7445 r^2 + 2.511 r - 1.674, \]  \quad (34)

is the fractional conversion of nitrogen, hydrogen feed at 250 atm. and 227 k.

The exact roots of (34) are

\[ \zeta_1 = 3.9485 + 0.3161i, \]
\[ \zeta_2 = 3.9485 - 0.3161i, \]
\[ \zeta_3 = -0.3841, \]
\[ \zeta_4 = 0.2778. \]  \quad (35)

The initial calculated values of (34) have been taken as follows:

\[ r_1^0(0) = 3.5 + 0.3i, \]
\[ r_2^0(0) = 3.5 - 0.3i, \]
\[ r_3^0(0) = -0.3 + 0.01i, \]
\[ r_4^0(0) = 1.8 + 0.01i. \]  \quad (36)

Table 2 clearly shows the dominance behavior of IWM2 over WDK2 iterative method in terms of CPU time in seconds and absolute error on same number of iterations \( k \) for nonlinear function. \( f_3(r) \).

Example 2 (see [6]). Van der Waal’s Fluid Model.

A Van der Waals fluid is the one which satisfies the equation of state:

\[ p = \frac{R \theta}{\nu - b} - \frac{a}{\nu^2}, \]  \quad (37)

where \( \theta \) is the fractional conversion of nitrogen, hydrogen feed at 250 atm. and 227 k.
Example 1 Example 2 Example 3 Example 4

Figure 1: (a), (b), (c), and (d) show basins of attraction for nonlinear function \( f_1(r) = r^3 + r - 40 \) of the iterative methods WDK, IWDK, WDK2, and IWM2 respectively.

Figure 2: (a), (b), (c), and (d) show basins of attraction for nonlinear function \( f_2(r) = \sin((r-1)/2)\cos((r-3)/2) + 1 \) of the iterative methods WDK, IWDK, WDK2, and IWM2, respectively.

Figure 3: Computational time in seconds of WDK2 and IWM2 for nonlinear function \( f_3(r) = f_6(r) \), respectively.

Table 2: Simultaneous finding of all roots.

| Method          | \( \epsilon_1^{(6)} \) | \( \epsilon_2^{(6)} \) | \( \epsilon_3^{(6)} \) | \( \epsilon_4^{(6)} \) |
|-----------------|-------------------------|-------------------------|-------------------------|-------------------------|
| WDK2            | 0.0                     | 0.0                     | 6.8e – 66               | 6.8e – 66               |
| IWM2            | 0.0                     | 0.0                     | 1.2e – 89               | 2.4e – 86               |

\[
18r^3 + 13r^2 + 9r - 3 = 0 \quad (39)
\]

or

\[
f_4(r) = 18r^3 + 13r^2 + 9r - 3. \quad (40)
\]

The exact roots of (40) are

\[
\zeta_1 = -0.476763 - 0.702381i,
\zeta_2 = -0.476763 + 0.702381i,
\zeta_3 = 0.2313104. \quad (41)
\]

The initial calculated values of (40) have been taken as follows:

\[
\begin{align*}
\tilde{r}_1(0) &= -0.4 - 0.7i, \\
\tilde{r}_2(0) &= -0.4 - 0.7i, \\
\tilde{r}_3(0) &= 0.2.
\end{align*} \quad (42)
\]

where \( R, a, \) and \( b \) are positive constants, \( P \) is the pressure, \( \theta \) is the absolute temperature, and \( v \) volume. We obtain a nonlinear equation

\[
\left( P + \frac{3}{v} \right)(3v - 1) = 8T, \quad (38)
\]

by setting \( P = (27b^2/p(a)), T = (27b\theta/8a), \) and \( r = (v/3b) \) Taking \( P = 6 \) and \( T = 2 \) in (37), we have
seconds and absolute error on the same number of iterations $k$ for nonlinear function $f_4(r)$.

**Example 3** (see [27]). Continuous Stirred Tank Reactor (CSTR).

An isothermal stirred tank reactor (CSTR) is considered here. Items A and R are fed to the reactor at rates of $Q$ and $q$, respectively. Complex reaction developed in the reactor is given as follows:

$$
A + R \rightarrow B,
B + R \rightarrow C,
C + R \rightarrow D,
C + R \rightarrow E.
$$

(43)

For a simple feedback control system, this problem was first tested by Douglas (see [28]). During his searching, he designed the following equation of transfer function of the reactor:

$$
H_c = \frac{2.98(r + 2.25)}{(r + 1.45)(r + 2.85)^2(r + 4.35)} = -1.
$$

(44)

$H_c$ being the gain of the proportional controller. This transfer function yields the following nonlinear equation by taking $H_c = 0$:

$$
f_5(r) = r^4 + 11.50r^3 + 47.49r^2 + 83.06325r + 51.23266875 = 0.
$$

(45)

The transfer function has the four negative real roots, i.e., $r_1 = -1.45, r_2 = -2.85, r_3 = -2.85,$ and $r_4 = -4.45$. The initial calculated values of (45) have been taken as follows:

$$
\begin{align*}
\tilde{r}_1(0) &= -1.0, \\
\tilde{r}_2(0) &= -1.1, \\
\tilde{r}_3(0) &= -2.2, \\
\tilde{r}_4(0) &= -3.9.
\end{align*}
$$

(46)

Table 4 clearly shows the dominance behavior of IWM2 over the WDK2 iterative method in terms of CPU time in seconds and absolute error on same number of iterations $k$ for nonlinear function $f_5(r)$.

**Example 4** (see [16]). Predator-Prey Model.

Consider the Predator-Prey model in which the predation rate is denoted by

$$
P(r) = \frac{kr^3}{a^3 + r^3}, \quad a, k > 0,
$$

(47)

where $r$ is the number of aphids as preys [6] and lady bugs as a predator. Obeying the Mathusian Model, the growth rate of aphids is defined as $G(r) = r_1^*r, r_2^* > 0$. To find the solution of the problem, we take the aphid density for which $P(r) = G(r)$ implies

$$
r_1^*r^3 - kr^2 + r^3a^3 = 0.
$$

(48)

Taking $k = 30$ (aphids eaten rate), $a = 20$ (number of aphids), and $r_1^* = 2^{-1/3}$ (rate per hour) in (48), we obtain

| Method | $e_1^{(3)}$ | $e_2^{(3)}$ | $e_3^{(3)}$ | $e_4^{(3)}$ |
|--------|-------------|-------------|-------------|-------------|
| WDK2   | 8036.0      | 8036.0      | 20.2        |             |
| IWM2   | 4.9e-97     | 4.9e-97     | 1.7e-100    | 0.004       |

Table 3: Simultaneous finding of all roots.

| Method | $e_1^{(3)}$ | $e_2^{(3)}$ | $e_3^{(3)}$ | $e_4^{(3)}$ |
|--------|-------------|-------------|-------------|-------------|
| WDK2   | 0.2         | 0.4         | 0.5         | 0.7         |
| IWM2   | 4.8e-37     | 9.4e-36     | 0.001       | 0.004       |

Table 4: Simultaneous finding of all roots.
The exact roots of (49) are
\[ \zeta_1 = 25.198, \quad \zeta_2 = 25.198, \quad \zeta_3 = 12.84. \]  
(50)

The initial estimates for \( f_6(r) \) has been taken as follows:
\[ r_1^0(0) = 1.8 + 8.7i, \]
\[ r_2^0(0) = 1.8 - 8.7i, \]  
(51)
\[ r_3^0(0) = 0.1 + 0.1i. \]

Table 5 clearly shows the dominance behavior of IWM2 over WDK2 iterative method in terms of CPU time in seconds and absolute error on the same number of iterations \( k \) for nonlinear function \( f_6(r) \).

4. Conclusion

In this work, new two-step derivative-free inverse iterative methods of convergence order 4 for the simultaneous approximations of all roots of a nonlinear equation (1) are introduced and discussed. Dynamical planes and basins of attraction are presented to show the global convergence behavior of inverse simultaneous iterative methods and two-step classical Weierstrass method. Brightness in color in the dynamical planes of IWM2 shows less number of iteration steps as compared to classical simultaneous methods WDK2 for finding all roots of (1). The results of numerical test examples from Tables 2–5, CPU time from Figure 3, and residual error from Figures 4–7, corroborate with theoretical analysis and illustrate the effectiveness and rapid convergence of our proposed derivative-free inverse simultaneous iterative method as compared to the WDK2.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this article.

Authors’ Contributions

All authors contributed equally in the preparation of this manuscript.

References

[1] H. T. Kung and J. F. Traub, “Optimal order of one-point and multipoint iteration,” Journal of the ACM, vol. 21, no. 4, pp. 643–651, 1974.
[2] G. H. Nedzhibov, “Inverse Weierstrass-Durand-Kerner iterative method,” International Journal of Applied Mathematics, vol. 28, no. 2, pp. 1258–1264, 2013.
[3] P. D. Proinov and M. T. Vasileva, “On a family of Weierstrass-type root-finding methods with accelerated convergence,” Applied Mathematics and Computation, vol. 273, pp. 957–968, 2016.
[4] S. Kanno, N. Kjurkchiev, and T. Yamamoto, “On some methods for the simultaneous determination of polynomial zeros,” Japan Journal of Industrial and Applied Mathematics, vol. 13, pp. 267–288, 1995.
[5] P. D. Proinov and S. I. Ivanov, “Convergence analysis of Sakurai-Torii-Sugiura iterative method for simultaneous approximation of polynomial zeros,” Journal of Computational and Applied Mathematics, vol. 357, pp. 56–70, 2019.
[6] S. C. Chapra, Applied Numerical Methods with MATLAB® for Engineers and Scientists, McGraw-Hill, New York, NY, USA, 6th edition, 2010.
[7] N. A. Ivanov, R. Muneer, and I. Jabeen, “Some families of two-step simultaneous methods for determining zeros of nonlinear equations,” ISRN Applied Mathematics, vol. 2011, pp. 1–11, 2011.
[8] A. W. M. Nourein, “An improvement on two iteration methods for simultaneously determination of the zeros of a polynomial,” *International Journal of Computer Mathematics*, vol. 6, pp. 241–252, 1977.

[9] O. Aberth, “Iteration methods for finding all zeros of a polynomial simultaneously,” *Mathematics of Computation*, vol. 27, no. 122, p. 339, 1973.

[10] M. Cosnard and P. Fraigniaud, “Finding the roots of a polynomial on an MIMD multicomputer,” *Parallel Computing*, vol. 15, no. 1–3, pp. 75–85, 1990.

[11] Y. M. Chu1, N. Rafiq, M. Shams, S. Akram, N. A. Mir, and H. Kalsoom, “Computer methodologies for the comparison of some efficient derivative free simultaneous iterative methods for finding roots of non-linear equations,” *Computers, Materials & Continua*, vol. 66, no. 1, pp. 25–290, 2021.

[12] V. K. Kyncheva, V. V. Yotov, and S. I. Ivanov, “Convergence of Newton, Halley and Chebyshev iterative methods as methods for simultaneous determination of multiple polynomial zeros,” *Applied Numerical Mathematics*, vol. 112, pp. 146–154, 2017.

[13] M. R. Farmer, *Computing the Zeros of Polynomials Using the Divide and Conquer Approach*, Ph.D Thesis, Department of Computer Science and Information Systems, Birkbeck, University of London, WC1, London, England, 2014.

[14] S. I. Cholakov and M. T. Vasileva, “A convergence analysis of a fourth-order method for computing all zeros of a polynomial simultaneously,” *Journal of Computational and Applied Mathematics*, vol. 321, pp. 270–283, 2017.

[15] P. D. Ivanov and M. D. Petkova, “Convergence of the two-point Weierstrass root-finding method,” *Japan Journal of Industrial and Applied Mathematics*, vol. 31, no. 2, pp. 279–292, 2014.

[16] Y. M. Chu1, N. Rafiq, M. Shams, S. Akram, N. A. Mir, and H. Kalsoom, “Computer methodologies for the comparison of some efficient derivative free simultaneous iterative methods for finding roots of non-linear equations,” *Computers, Materials & Continua*, vol. 66, no. 1, pp. 25–290, 2021.

[17] V. K. Kyncheva, V. V. Yotov, and S. I. Ivanov, “Convergence of Newton, Halley and Chebyshev iterative methods as methods for simultaneous determination of multiple polynomial zeros,” *Applied Numerical Mathematics*, vol. 112, pp. 146–154, 2017.

[18] M. R. Farmer, *Computing the Zeros of Polynomials Using the Divide and Conquer Approach*, Ph.D Thesis, Department of Computer Science and Information Systems, Birkbeck, University of London, WC1, London, England, 2014.

[19] S. I. Cholakov and M. T. Vasileva, “A convergence analysis of a fourth-order method for computing all zeros of a polynomial simultaneously,” *Journal of Computational and Applied Mathematics*, vol. 321, pp. 270–283, 2017.

[20] P. D. Ivanov and M. D. Petkova, “Convergence of the two-point Weierstrass root-finding method,” *Japan Journal of Industrial and Applied Mathematics*, vol. 31, no. 2, pp. 279–292, 2014.

[21] Y. M. Chu1, N. Rafiq, M. Shams, S. Akram, N. A. Mir, and H. Kalsoom, “Computer methodologies for the comparison of some efficient derivative free simultaneous iterative methods for finding roots of non-linear equations,” *Computers, Materials & Continua*, vol. 66, no. 1, pp. 25–290, 2021.

[22] V. K. Kyncheva, V. V. Yotov, and S. I. Ivanov, “Convergence of Newton, Halley and Chebyshev iterative methods as methods for simultaneous determination of multiple polynomial zeros,” *Applied Numerical Mathematics*, vol. 112, pp. 146–154, 2017.

[23] M. R. Farmer, *Computing the Zeros of Polynomials Using the Divide and Conquer Approach*, Ph.D Thesis, Department of Computer Science and Information Systems, Birkbeck, University of London, WC1, London, England, 2014.

[24] S. I. Cholakov and M. T. Vasileva, “A convergence analysis of a fourth-order method for computing all zeros of a polynomial simultaneously,” *Journal of Computational and Applied Mathematics*, vol. 321, pp. 270–283, 2017.