Research Article
On Dynamics of Iterative Techniques for Nonlinear Equation with Applications in Engineering

Mudassir Shams, Nazir Ahmad Mir, Naila Rafiq, A. Othman Almatroud, and Saima Akram

Department of Mathematics and Statistics, Riphah International University I-14, Islamabad 44000, Pakistan
Department of Mathematics, NUML, Islamabad, Pakistan
Department of Mathematics, Faculty of Science, University of Ha’il, Ha’il, Saudi Arabia
Centre for Advanced Studies in Pure and Applied Mathematics, Bahauddin Zakariya University, Multan, Pakistan

Correspondence should be addressed to Mudassir Shams; shams6262@gmail.com

1. Introduction
To solve a nonlinear equation,

\[ f(s) = 0 \] (1)

is the oldest problem of science, in general, and mathematics, in particular. The nonlinear equations have diverse applications in many areas of science and engineering. In general, to find the roots of (1), we look towards iterative schemes, which can be further classified as to approximate the single root and all roots of (1). In this article, we are going to work on both types of iterative methods. Many iterative methods of different convergence orders already exist in the literature (see [1–11]) to approximate the roots of (1). Ostrowski [12] defined the efficiency index \( I \) to classify these iterative methods in terms of their convergence order \( k \) and number of function evaluations or its derivatives per iteration, say \( v \), i.e.,

\[ I = k^{1/v}. \] (2)

An iterative method is said to be optimal according to the Kung–Traub conjecture [13], if

\[ H = 2^{k-1}. \] (3)

holds, where \( H \) is the optimal order of convergence. The aforementioned methods are used to approximate one root at a time. But, mathematicians are also interested in finding all the roots of (1) simultaneously. This is due to the fact that simultaneous iterative methods are very popular due to their wider region of convergence, are more stable as compared to single root finding methods, and can be implemented for parallel computing as well. More detail on single as well as simultaneous determination of all roots can be found in [1, 10–28] and the reference cited therein.

The most famous of the single root finding methods is the classical Newton–Raphson method:
\[ s_{i+1} = s_i - \frac{f(s_i)}{f'(s_i)} \quad i = 1, 2, \ldots \tag{4} \]

Method (4) is optimal with an efficiency of 1.41 by the Traub conjecture. If we use Weierstrass’ correction [26]:

\[ \frac{f(s_i)}{f'(s_i)} = \frac{f(s_i)}{\prod_{j=1}^{n} (s - s_j)} \quad (5) \]

in (4), then, we get classical Weierstrass–Dochev methods to approximate all roots of nonlinear equation (1) as

\[ s_{i+1} = s_i - \frac{f(s_i)}{\prod_{j=1}^{n} (s - s_j)} \quad i, j = 1, 2, 3, \ldots, n. \tag{6} \]

Method (6) has a convergence order 2. Later, Aberth [29] presented the 3rd order simultaneous method, given as

\[ s_{i+1} = s_i - \frac{1}{(1/N(s_i)) - \sum_{j \neq i} (1/(s_i - s_j))} \tag{7} \]

where \( N(s_i) = (f'(s_i)/f(s_i)) \).

First of all, we construct a family of optimal fourth-order methods using the procedure of weight function and then convert it into simultaneous iterative methods for finding all distinct as well as multiple roots of nonlinear equation (1).

### 2. Construction of Methods and Convergence Analysis

King [9] presented the following optimal fourth-order method (abbreviated as E1):

\[
\begin{align*}
  y_i &= s_i - \frac{f(s_i)}{f'(s_i)}, \\
  z_i &= y_i - \frac{f(s_i) + \beta f(y_i)}{f'(s_i)} \left( f(y_i) \right)
\end{align*}
\tag{8}
\]

Cordero et al. [7] gave the fourth-order optimal method as follows (abbreviated as E2):

\[
\begin{align*}
  y_i &= s_i - \frac{f(s_i)}{f'(s_i)}, \\
  z_i &= s_i - \frac{f(s_i) + f(y_i)}{f'(s_i)} - \frac{(f(y_i))^2 (2 f(s_i) + f(y_i))}{(f(s_i))^2 f'(s_i)}
\end{align*}
\tag{9}
\]

Chun [4] in 2008 gave the fourth-order optimal method as follows (abbreviated as E3):

\[
\begin{align*}
  y_i &= s_i - \left( f(s_i) \right), \\
  z_i &= s_i - \left( f(s_i) \right) \left( 1 + \frac{f(y_i)}{f(s_i)} + \frac{f(y_i)^2}{f'(s_i)} \right)
\end{align*}
\tag{10}
\]

Maheshwari [11] gave the fourth-order optimal method as follows (abbreviated as E4):

\[
\begin{align*}
  y_i &= s_i - \left( f(s_i) \right), \\
  z_i &= s_i - \frac{f(s_i)^2}{f'(s_i)} \left( f(y_i) - f(s_i) \right)
\end{align*}
\tag{11}
\]

Chun [3] gave the fourth-order optimal method as follows (abbreviated as E5):

\[
\begin{align*}
  y_i &= s_i - \left( f(s_i) \right), \\
  z_i &= s_i - \frac{(f(s_i))^2 + (f(y_i))^2}{(f'(s_i))(f(s_i) - f(y_i))}
\end{align*}
\tag{12}
\]

Kou et al. [10] gave the fourth-order optimal method as follows (abbreviated as E6):

\[
\begin{align*}
  y_i &= s_i - \left( f(s_i) \right), \\
  z_i &= s_i - \frac{f(s_i)^2 (f(y_i))^2}{f'(s_i)}
\end{align*}
\tag{13}
\]

Behzad [5] give the fourth-order optimal method as follows (abbreviated as E7):

\[
\begin{align*}
  y_i &= s_i - \left( f(s_i) \right), \\
  z_i &= s_i - \frac{(f(s_i) + f(y_i))^2 (f(y_i))}{(f'(s_i))(f(s_i) - f(y_i))}
\end{align*}
\tag{14}
\]

Chun [2] in 2006 gave the fourth-order optimal method as follows (abbreviated as E8):

\[
\begin{align*}
  y_i &= s_i - \left( f(s_i) \right), \\
  z_i &= s_i - \frac{(f(s_i) + f(y_i))}{(f'(s_i))(f(s_i) - f(y_i))}
\end{align*}
\tag{15}
\]

Ostrowski [12] give the fourth-order optimal method as follows (abbreviated as E9):
Mathematical Problems in Engineering

Theorem 1. Let \( \zeta \in I \) be a simple root of a sufficiently differentiable able function \( f \): \( I \subseteq \mathbb{R} \rightarrow \mathbb{R} \) in an open interval \( I \). If \( s_0 \) is sufficiently close to \( \zeta \) and \( \Psi (u) \) is a real-valued function satisfying \( \Psi (0) = 1, \Psi' (0) = (2/3), \) and \( \Psi'' (0) < \infty \), then the convergence order of the family of iterative method (17) is four and the error equation is given by

\[
e_{r+1} = \left( \frac{11}{3} c_2^3 - c_2 c_3 - \frac{3}{2} c_2^2 \Psi'' (0) \right) e_i^4 + O(e_i^5),
\]

where \( c_m = (f^m(\zeta)/m!) / f' (\zeta) \), \( m \geq 2 \).

Proof. Let \( \zeta \) be a simple root of \( f \) and \( s_i = \zeta + e_i \). By Taylor’s series expansion of \( f (s_i) \) around \( s = \zeta \), taking \( f (\zeta) = 0 \), we get

\[
f (s_i) = f' (\zeta) (e_i + c_2 e_i^2 + c_3 e_i^3 + c_4 e_i^4 + O(e_i^5)),
\]

and

\[
f' (s_i) = f' (\zeta) \left(1 + 2c_2 e_i + 3c_3 e_i^2 + 4c_4 e_i^3 + O(e_i^4)\right).
\]

Dividing (19) by (20), we have

\[
f (s_i) = e - c_2 e_i^2 + (2c_2 + 2c_3) e_i^3 + O(e_i^4),
\]

so

\[
y_i = \zeta + c_2 e_i^2 + (-2c_2^2 + 2c_3) e_i^3 + \cdots,
\]

\[
f (y_i) = c_2 e_i^2 + (-4c_2^2 + 2c_3) e_i^3 + \cdots.
\]

Now,

\[
\left( \frac{f (y_i)}{f' (s_i)} \right) = c_2 e_i^2 + (-4c_2^2 + 2c_3) e_i^3 + \cdots,
\]

\[
\left( \frac{f (y_i)}{f' (s_i)} \right) = u = c_2 e_i + (-3c_2^2 + 2c_3) e_i^2 + \cdots.
\]

Expanding \( \Psi (u) \) about the origin, we have

\[
\Psi (u) = \Psi (0) + \Psi' (0) u + \frac{1}{2} \Psi'' (0) u^2 + \cdots
\]

\[
= \Psi (0) + \Psi' (0) c_2 e_i + \left(-3 \Psi' (0) c_2^2 + 2 \Psi'' (0) c_2 + \frac{1}{2} \Psi''' (0) c_2^3\right) e_i^3 + \cdots,
\]

\[
\beta (f (y_i))^2 + (\Psi (u))^3 = \Psi^3 (0) + 3 \Psi' (0) \Psi^2 (0) c_2 e_i + \cdots,
\]

\[
e_{r+1} = \left(-\Psi^3 (0) c_2 + c_3\right) e_i^2 + \left(-3 \Psi (0) \Psi' (0) c_2^2 + 4 \Psi^3 (0) c_2^3 - 2 \Psi^3 (0) c_2 - 2c_2^2 + 2c_3\right) e_i^3 + \cdots.
\]

By putting \( \Psi (0) = 1 \) and \( \Psi' (0) = (2/3) \) in (25), we have

\[
e_{r+1} = \left(\frac{11}{3} c_2^3 - c_2 c_3 - \frac{3}{2} c_2^2 \Psi'' (0) \right) e_i^4 + O(e_i^5).
\]

Hence, it proves the theorem.

2.1. The Concrete Fourth-Order Methods. We now construct some concrete forms of the family of methods described by (17). Let us take the function \( \Psi (u) \) satisfying the conditions of Theorem 1.
Therefore, we get the following new three iterative methods with arbitrary constant $\beta \in \mathbb{R}$ and by choosing different weight functions given in Table 1:

Concrete method 1 (abbreviated as Q1):

\[
\begin{align*}
    y_i &= s_i - \left( \frac{f(s_i)}{f'(s_i)} \right), \\
    z_i &= y_i - \left( \frac{f(y_i)}{f'(s_i)} \right) \left[ \beta (f(y_i))^2 + \left( 1 + \frac{3}{2} u \right)^3 \right]. \\
\end{align*}
\]

Concrete method 2 (abbreviated as Q2):

\[
\begin{align*}
    y_i &= s_i - \left( \frac{f(s_i)}{f'(s_i)} \right), \\
    z_i &= y_i - \left( \frac{f(y_i)}{f'(s_i)} \right) \left[ \beta (f(y_i))^2 + \left( 1 + \frac{3}{2} u + au^2 \right)^3 \right]. \\
\end{align*}
\]

Concrete method 3 (abbreviated as Q3):

\[
\begin{align*}
    y_i &= s_i - \left( \frac{f(s_i)}{f'(s_i)} \right), \\
    z_i &= y_i - \left( \frac{f(y_i)}{f'(s_i)} \right) \left[ \beta (f(y_i))^2 + \left( \frac{1}{1 - 23m} \right)^3 \right]. \\
\end{align*}
\]

where $u = (f(y_i)/f(s_i))$ and $\beta \in \mathbb{R}$.

<table>
<thead>
<tr>
<th>S. no.</th>
<th>$\Psi (u), u = (f(y_i)/f(s_i)), \Psi (0) = 1, \Psi' (0) = (2/3)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\Psi (u) = 1 + (2/3)u$</td>
</tr>
<tr>
<td>2</td>
<td>$\Psi (u) = 1 + (2/3)u + au^2$</td>
</tr>
<tr>
<td>3</td>
<td>$\Psi (u) = (1 - (2/3)u)$</td>
</tr>
</tbody>
</table>

Note. $u = (f(y_i)/f(s_i))$ and $a \in \mathbb{R}$.

set denoted by $F(R)$. The iterative methods when applied to find the roots of (1) provide the rational map. But, we are interested in the basins of attraction of the roots of nonlinear function (1). Fatou set $F(Q)$ contains the basins of attraction of different roots is a well-known fact. In general, the Julia set (a fractal and rational map) behaves as unstable in this region. For the dynamical and graphical point of view, we take $2000 \times 2000$ grids of square $[-2.5, 2.5]^2 \in C^2$. To each root of (1), we assign a color to which the corresponding orbit of the iterative method starts and converges to a fixed point. Take color map as Jet. We use $|s_{i+1} - s_i| < 10^{-3}$ as a stopping criteria, and the maximum number of iterations is taken as 20. We mark a dark blue point, if the orbit of the iterative method does not converge to the root after 20 iterations which means it has a distance greater than to any other root. Different color is used for different roots. Iterative methods have different basins of attraction distinguished by their colors. In basins, brightness in color represents the number of iterations to achieve the root of (1). Note that the darkest blue regions denote the lack of convergence to any root of (1). Finally, in Tables 2–5, we present the elapsed time of basins of attraction corresponding to iterative maps (Q1–Q3 and E1–E9) using the tic-toc command in code using MATLAB (R2011b). Figures 1–4 show the basins of attraction of iterative methods (Q1–Q3 and E1–E9) for nonlinear functions $f_1(s) = s^3 + s - 1$, $f_2(s) = e^{-s} - 1$, $f_3(s) = ((s^2 + 2.87s^2 - 10.28)/4.62) - s$, and $f_4(s) = \sin((s + 2)/2)\sin((s - 3)/3)$, respectively. By observing the basins of attraction, we can easily judge the stability of iterative methods (Q1–Q3 and E1–E9). Elapsed time, divergent regions, and brightness in color presents that Q1–Q3 is better than E1–E9.

2.2. Complex Dynamical Study of Families of Iterative Methods. Here, we discuss the dynamical study of iterative methods (Q1–Q3 and E1–E9). We investigate the regions from where we take the initial estimates to achieve the roots of the nonlinear equation. We actually numerically approximate the domain of attractions of the roots as a qualitative measure, that is, how the iterative methods depend on the choice of initial estimates? To answer this question on the dynamical behavior of the iterative methods, we investigate the dynamics of the methods (Q1–Q3) and compare it with (E1–E9). Let us recall some basic concepts of this study in the background context of complex dynamics. For more details on the dynamical behavior of the iterative methods, one can consult [6, 30, 31]. Taking a rational function $R : C \rightarrow C$, where $C$ denotes the complex plane, the orbit $s_0 \in C$ is defined as a set such as $\text{orb}(s) = \{s_0, R(s_0), R^2(s_0), R^3(s_0), \ldots, R^n(s_0), \ldots \}$. The convergence of $\text{orb}(s) \rightarrow s^*$ is understood in the sense, if $\lim_{n \rightarrow \infty} R^n(s) = s^*$. A point $s_0 \in C$ is known as periodic with a minimal period $m$ if $R^m(s_0) = s_0$ holds, where $m$ is the smallest positive integer. A periodic point for $m = 1$ is known as fixed, attracting if $|R^k(s)| < 1$, repelling if $|R^k(s)| > 1$, and neutral otherwise. An attracting point $s_0 \in C$ defines the basin of attraction as the set of starting points whose orbit tends to $s^*$. The closure of the set of its repelling periodic points of a rational map is known as the Julia set denoted by $J(R)$, and its complement is the Fatou
Figures 4(a)–4(l) present the basins of attraction of methods (Q1–Q3 and E1–E9) for a nonlinear function $f_4(s) = \sinh \left( \frac{s+2}{2} \right) \sinh \left( \frac{s-3}{3} \right)$. In Figures 4(a)–4(l), brightness of the color in basins shows the less number of iterations for convergence of iterative methods. Table 5 shows the elapsed time of Q1–Q3 and E1–E9.

### 3. Generalization to Simultaneous Methods

Suppose nonlinear equation (1) has $n$ roots. Then, $f(s)$ and $f'(s)$ can be approximated as

$$f(s) = \prod_{j=1}^{n} (s - s_j), \quad f'(s) = \sum_{j=1}^{n} \prod_{k=1 \atop k \neq j}^{n} (s - s_j).$$

This implies

$$\frac{f'(s)}{f(s)} = \sum_{j=1}^{n} \left( \frac{1}{s - s_j} \right) = \frac{1}{(1/(s - s_j)) - \sum_{j=1}^{n} (1/(s - s_j))}.$$

(30)

This gives the Aberth–Ehrlich method (7):

$$s_{i+1} = s_i - \frac{1}{(1/N(s_i)) - \sum_{j=1}^{n} (1/(s_i - s_j))} \text{ where } N(s_i) = \frac{f(s_i)}{f'(s_i)}.$$

(32)

Now from (31), an approximation of $(f(s_i)/f'(s_i))$ is formed by replacing $s_j$ with $z_{ij}$ as follows:

$$\frac{f(s_i)}{f'(s_i)} = \frac{1}{(1/N(s_i)) - \sum_{j=1}^{n} (1/(s_i - z_{ij}))}, \quad t = 1, 2, 3.$$

(33)

Using (33) in (4), we have the following method for finding all the distinct roots:

$$s_{i+1} = s_i - \frac{1}{(1/N(s_i)) - \sum_{j=1}^{n} (1/(s_i - z_{ij}))}.$$

(34)

In case of multiple roots, we have the following method:

$$s_{i+1} = s_i - \frac{1}{(1/N(s_i)) - \sum_{j=1}^{n} \sigma_j (1/(s_i - z_{ij}))}, \quad t = 1, 2, 3.$$

(35)
Figure 1: Basins of attraction of iterative methods (Q1–Q3 and E1–E9) for nonlinear function $f_1(s)$. 
Figure 2: Basins of attraction of iterative methods (Q1–Q3 and E1–E9) for nonlinear function $f_2(s)$. 

Mathematical Problems in Engineering 7
Figure 3: Basins of attraction of iterative methods (Q1–Q3 and E1–E9) for nonlinear function $f_3(s)$. 
Figure 4: Basins of attraction of iterative methods (Q1–Q3 and E1–E9) for nonlinear function $f_4(s)$. 
where $\sigma_i$ is the multiplicity of the root $s_i$ and

$$
z_{1j} = y_j - \left( \frac{f(y_j)}{f(s_j)} \right) \left( \beta(f(y_j))^2 + \left(1 + \frac{2}{3}u^* \right)^3 \right),$$

$$
z_{2j} = y_j - \left( \frac{f(y_j)}{f(s_j)} \right) \left( \beta(f(y_j))^2 + \left(1 + \frac{2}{3}u^* + \alpha(u^*)^2 \right)^3 \right),$$

$$
z_{3j} = y_j - \left( \frac{f(y_j)}{f(s_j)} \right) \left( \beta(f(y_j))^2 + \left(\frac{1}{1 - (2/3)u^*} \right)^3 \right),$$

(36)

in which $y_j = s_j - (f(y_j)/f^e(s_j))$ and $u^* = (f(y_j)/f(s_j))$. Using correction $z_{ij}$, we get the following new family of simultaneous iterative methods for extracting multiple roots of nonlinear equation (1):

$$s_{i+1} = s_i - \sum_{j=1}^{n} \frac{\sigma_i}{N(s_j)} \frac{\sigma_i}{(s_i - z_{ij})^2} - \sum_{j \neq i}^{n} \frac{\sigma_j}{(s_i - z_{ij})},$$

(37)

Thus, we have constructed new three simultaneous iterative methods (37) abbreviated as SM1–SM3.

3.1. Convergence Analysis. In this section, we discuss the convergence analysis of a family of simultaneous methods (SM1–SM3) which is given in form of the following theorem. Obviously, convergence for the method (34) will follow from the convergence of the method (SM1–SM3) from Theorem 2 when the multiplicities of the roots are one.

**Theorem 2** Let $\zeta_1, \zeta_2, \ldots, \zeta_n$ be the $n$ number of simple roots of nonlinear equation (1). If $s_1(0), s_2(0), s_3(0), \ldots, s_n(0)$ be the initial approximations of the roots, respectively, and sufficiently close to actual roots, then the order of convergence of the method (SM1–SM3) equals six.

**Proof.** Let $e_i = s_i - \zeta_i$ and $e_i' = s_{i+1} - \zeta_i$ be the errors in $s_i$ and $s_{i+1}$ approximations, respectively. Then, obviously for distinct roots

$$
\frac{1}{N(s_i)} = \left( \frac{f(s_i)}{f^e(s_i)} \right) = \frac{1}{(s_i - \zeta_i)} + \frac{1}{(s_i - \zeta_i)} + \cdots + \frac{1}{(s_i - \zeta_i)}.
$$

(38)

Thus, for multiple roots, we have from (37) that

$$s_{i+1} - \zeta_i = s_i - \zeta_i - \left( \frac{\sigma_i}{(s_i - \zeta_i)} + \sum_{j \neq i}^{n} \frac{\sigma_j}{(s_i - z_{ij})}\right),$$

$$e_i = e_i - \frac{\sigma_i}{\sigma_i + e_i} \sum_{j \neq i}^{n} \frac{\sigma_j}{(s_i - z_{ij})},$$

(39)

where $z_{ij} - \zeta_j = \epsilon_j$ from (18) and $E_i = (-\sigma_j/(s_i - \zeta_j))$. Thus,$E_i = (-\sigma_j/(s_i - \zeta_j))$. Thus,

$$
e_i^2 = \sum_{j=1}^{n} (E_i)^2,$$

$$\epsilon_i = \sum_{j=1}^{n} (E_i)^2$$

(40)

If it is assumed that absolute values of all errors $e_j$ ($j = 1, 2, 3, \ldots$) are of the same order, say $|e_j| = O(e)$, then from (40), we have

$$\epsilon_i^2 = O(e_i)^6.$$

(41)

Hence, the theorem is proved.
4. Computational Aspect

Here, we compare the computational efficiencies of the Petkovic et al. [21] method and the new methods (SM1–SM3). As presented in [21], the efficiency of an iterative method can be estimated using the efficiency index given by

\[ EL(m) = \frac{\log u}{D}, \]  

where \( D \) is the computational cost and \( u \) is the order of convergence of the iterative method. Arithmetic operations per iteration with certain weight depending on the execution time of operation are used to evaluate the computational cost \( D \). The weights used for division, multiplication, and addition plus subtraction are \( w_d, w_m, \) and \( w_{AS}, \) respectively. For a given polynomial of degree \( m \) and \( n \) roots, the number of division, multiplication, and addition and subtraction per iteration for all roots are denoted by \( D_m, M_m, \) and \( AS_m. \) The cost of computation can be calculated as

\[ D = D(m) = w_d AS_m + w_m M_m + w_d D_m. \]

Thus, (42) becomes

\[ EL(m) = \frac{\log u}{w_d AS_m + w_m M_m + w_d D_m}. \]

Consider that the number of operations of a complex polynomial with real and complex roots reduces to the operation of real arithmetic, which is given in Table 6 as a polynomial degree \( m \) taking the dominant term of order \( m^6 \). Applying (44) and the data given in Table 6, we calculate the percentage ratio \( \rho((SM1–SM3), (PJ6)) [21] \) given by

\[
\begin{align*}
\rho((SM1–SM3), (PJ6)) &= \left( \frac{EL(SM1–SM3)}{EL(PJ6)} - 1 \right) \times 100, \\
\rho((PJ6), (SM1–SM3)) &= \left( \frac{EL(PJ6)}{EL(SM1–SM3)} - 1 \right) \times 100,
\end{align*}
\]

where PJ6 is the Petkovic method [21] of order 4. Figures 5(a)–5(d) graphically illustrate these percentage ratios. Figures 5(a)–5(c) show the computational efficiency of methods (SM1–SM3) w.r.t the method PJ6, and Figure 5(d) shows the computational efficiency of the method PJ6 w.r.t (SM1–SM3). It is evident from Figures 5(a)–5(d) that the newly constructed simultaneous methods (SM1–SM3) are more efficient as compared to PJ6 [21].

5. Numerical Results

Here, some numerical examples are considered in order to demonstrate the performance of our family of one-step fourth-order single root finding methods (Q1–Q3) and sixth-order simultaneous methods (SM1–SM3), respectively. We compare our family of optimal fourth-order single root finding methods (Q1–Q3) with E1–E9 methods. The family of simultaneous methods (SM1–SM3) of order six is compared with the PJ6 method [21] of the same order. All the computations are performed using CAS Maple 18 with 9000 (64 digits floating point arithmetic in case of simultaneous methods) significant digits. For single root finding methods, the stopping criteria is as follows:

\[
\begin{align*}
(i) & \left| f'(s_i) \right| < \epsilon, \\
(ii) & \left| s_i^{(k)} - a \right| < \epsilon, 
\end{align*}
\]

whereas

\[ e_1 = \| f(s_i) \|_2, \]

for simultaneous methods. We take \( \epsilon = 10^{-600} \) for the single root finding method and \( \epsilon = 10^{-30} \) for simultaneous determination of all roots of nonlinear equation (1).

Numerical tests examples from [32–34] are provided in Tables 7–13. In Tables 7, 9, 11, and 12, we present the numerical results for simultaneous determination of all roots, while Tables 8, 10, and 13 represent for single root finding methods. In all tables, CO represents the convergence order; \( n \), the number of iterations; \( \rho \), the computational order of convergence; and CPU, the computational time in seconds. Table 14 shows the values of arbitrary parameters \( \alpha \) and \( \beta \) used in iterative methods Q1–Q3 for test Examples 1–3.

We also calculate the CPU execution time, as all the calculations are done using Maple 18 (Processor Intel(R) Core(TM) i3-3110m CPU@2.4GHz with 64-bit operating system). We observe from tables that CPU time of the methods SM1–SM3 is comparable or better than that of the PJ6 method [21], showing the efficiency of our methods (SM1–SM3) as compared to them.

6. Applications in Engineering

In this section, we consider two examples from engineering.

Example 1. (beam designing model, see [34]).

An engineer considers that a problem of embedment \( s \) of a sheet-pile wall results a nonlinear function given as

\[ f_s(s) = \frac{s^3 + 2.875s - 10.28}{4.62} - s. \]

The exact roots of (48) are \( \zeta_1 = 2.0021, \zeta_2 = -3.3304, \) and \( \zeta_3 = -1.5417 \) as shown in Figure 6.

The initial estimates for \( f_s(s) \) are taken as

<table>
<thead>
<tr>
<th>Methods</th>
<th>CO</th>
<th>AM</th>
<th>Mm</th>
<th>Dm</th>
</tr>
</thead>
<tbody>
<tr>
<td>SM1</td>
<td>6</td>
<td>6m^2 + O(m)</td>
<td>6m^2 + O(m)</td>
<td>2m^2 + O(m)</td>
</tr>
<tr>
<td>SM2</td>
<td>6</td>
<td>7m^2 + O(m)</td>
<td>5m^2 + O(m)</td>
<td>2m^2 + O(m)</td>
</tr>
<tr>
<td>SM3</td>
<td>6</td>
<td>6m^2 + O(m)</td>
<td>5m^2 + O(m)</td>
<td>2m^2 + O(m)</td>
</tr>
<tr>
<td>PJ6</td>
<td>6</td>
<td>8m^2 + O(m)</td>
<td>6m^2 + O(m)</td>
<td>2m^2 + O(m)</td>
</tr>
</tbody>
</table>
\[
\begin{align*}
\frac{dS}{dt} &= \frac{S}{S_0} - \frac{S^2}{S_0^2} - S^3 + \frac{S^4}{S_0^4}, \\
S &= \text{fractional conversion of nitrogen, with hydrogen feed at 250 atm. and 227 k.}
\end{align*}
\]

**Example 2** (fractional conversion, see [32]).

As the expression described in [14, 35],

\[
f_6(s) = s^4 - 7.79075s^3 + 14.7445s^2 + 2.511s - 1.674
\]

\[
f_5(s) = \left(\frac{s^3 + 2.875s^2 - 10.28}{4.62}\right) - s
\]

\[
\zeta_1 = 2.0021, \quad \zeta_2 = -3.3304, \quad \zeta_3 = -1.5417
\]

\((0) S_1 = 2.5, \quad (0) S_2 = -7.4641, \quad (0) S_4 = -0.5359
\]

Table 7: Simultaneously finding of all roots.

<table>
<thead>
<tr>
<th>Method</th>
<th>CO</th>
<th>CPU</th>
<th>n</th>
<th>e1</th>
<th>e2</th>
<th>e3</th>
</tr>
</thead>
<tbody>
<tr>
<td>PJ6</td>
<td>6</td>
<td>0.032</td>
<td>5</td>
<td>2.0e-40</td>
<td>1.1e-38</td>
<td>1.5e-39</td>
</tr>
<tr>
<td>SM1–SM3</td>
<td>6</td>
<td>0.015</td>
<td>5</td>
<td>0.0</td>
<td>3.2e-27</td>
<td>0.0</td>
</tr>
</tbody>
</table>

\[
(0) S_1 = 2.5, \\
(0) S_2 = -7.4641, \\
(0) S_3 = -0.5359.
\]

Table 8: Comparison of the optimal 4th order single root finding methods for \(f_5(s)\).

| Method | \(|s_i^{(0)} - s|\) | \(|f_i^{(0)}(s_i)|\) | CPU | \(\rho\) |
|--------|------------------|-----------------|-----|-----|
| Q1     | 2.0e-58          | 1.0e-76         | 0.062 | 4.0 |
| Q2     | 3.8e-63          | 2.0e-81         | 0.031 | 4.0 |
| Q3     | 2.1e-60          | 1.1e-78         | 0.031 | 4.0 |
| E1     | 1.6e-67          | 8.5e-86         | 0.062 | 4.0 |
| E2     | 4.8e-66          | 2.4e-84         | 0.031 | 4.0 |
| E3     | 1.8e-67          | 9.5e-86         | 0.031 | 4.0 |
| E4     | 3.6e-66          | 1.8e-84         | 0.063 | 4.0 |
| E5     | 3.2e-65          | 1.6e-83         | 0.531 | 4.0 |
| E6     | 3.1e-67          | 1.9e-85         | 0.0078 | 4.0 |
| E7     | 1.3e-64          | 7.1e-83         | 0.046 | 4.0 |
| E8     | 3.2e-65          | 1.6e-83         | 0.047 | 4.0 |
| E9     | 6.9e-69          | 3.5e-87         | 0.062 | 4.0 |

is the fractional conversion of nitrogen, with hydrogen feed at 250 atm. and 227 k. The exact roots of (50) are \(\zeta_1 = 3.9485 + 0.3161i, \quad \zeta_2 = 3.9485 - 0.3161i, \quad \zeta_3 = -0.3841, \quad \text{and} \quad \zeta_4 = 0.2778\). Real roots of (50) are shown in Figure 7. The initial estimates for \(f_6(s)\) are taken as
Table 9: Simultaneously finding of all roots of \( f_6(s) \).

\[
f_6(s) = s^4 - 7.7907s^3 + 2.511s - 1.674
\]
\[
\zeta_1 = 3.9485 + 0.3161i, \quad \zeta_2 = 3.9485 - 0.3161i, \quad \zeta_3 = -0.3841, \quad \zeta_4 = 0.2778
\]
\[
S_1^{(0)} = 3.5 + 0.3i, \quad S_2^{(0)} = 3.5 - 0.3i, \quad S_3^{(0)} = 0.3 + 0.01i, \quad S_4^{(0)} = 1.8 + 0.01i
\]

<table>
<thead>
<tr>
<th>Method</th>
<th>CO</th>
<th>CUP</th>
<th>( n )</th>
<th>( e1 )</th>
<th>( e2 )</th>
<th>( e3 )</th>
<th>( e4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>PJ6</td>
<td>6</td>
<td>0.031</td>
<td>4</td>
<td>4.6e-62</td>
<td>7.0e-62</td>
<td>3.7e-93</td>
<td>0.0</td>
</tr>
<tr>
<td>SM1–SM3</td>
<td>6</td>
<td>0.015</td>
<td>4</td>
<td>0.0</td>
<td>0.0</td>
<td>1.0e-137</td>
<td>1.0e-133</td>
</tr>
</tbody>
</table>

Table 10: Comparison of the optimal 4th order single root finding methods for \( f_6(s) \).

\[
f_6(S) = s^4 - 7.79075s^3 + 14.744s^2 + 2.5111s - 1.674, \quad s_0 = 0.27, \quad CO = 4
\]

| Method | \( |s_1^{(0)} - \alpha| \) | \( |f_1^{(0)}(s_1)| \) | CPU | \( \rho \) |
|--------|-----------------|-----------------|-----|--------|
| Q1     | 9.5e-1947       | 3.0e-7783       | 0.079 | 4.0    |
| Q2     | 1.3e-2120       | 2.9e-8477       | 0.074 | 4.0    |
| Q3     | 5.9e-2002       | 3.2e-8004       | 0.073 | 4.0    |
| E1     | 4.1e-2016       | 6.5e-8061       | 0.078 | 4.0    |
| E2     | 1.4e-1934       | 1.6e-7734       | 0.062 | 4.0    |
| E3     | 2.8e-2016       | 1.5e-8061       | 0.047 | 4.0    |
| E5     | 2.1e-1934       | 9.1e-7734       | 0.063 | 4.0    |
| E6     | 8.6e-1906       | 2.6e-7619       | 0.062 | 4.0    |
| E7     | 7.4e-1970       | 8.4e-7876       | 0.078 | 4.0    |
| E8     | 1.0e-1881       | 6.7e-7523       | 0.079 | 4.0    |
| E9     | 8.6e-1906       | 2.6e-7619       | 0.047 | 4.0    |

Table 11: Simultaneously finding of all roots for \( f_7(s) \).

\[
f_7(s) - \sin((s - 1)/2)\sin ((s - 2)/2)\sin ((s - 2.5)/2)
\]
\[
\zeta_1 = 1, \quad \zeta_2 = 2, \quad \zeta_3 = 2.5
\]
\[
S_1^{(0)} = -0.2, \quad S_2^{(0)} = 1.7, \quad S_3^{(0)} = 3
\]

<table>
<thead>
<tr>
<th>Method</th>
<th>CO</th>
<th>CUP</th>
<th>( n )</th>
<th>( e1 )</th>
<th>( e2 )</th>
<th>( e3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>PJ6</td>
<td>6</td>
<td>0.047</td>
<td>3</td>
<td>0.01</td>
<td>0.01</td>
<td>0.2e-5</td>
</tr>
<tr>
<td>SM1–SM3</td>
<td>6</td>
<td>0.031</td>
<td>3</td>
<td>8.8e-3</td>
<td>5.4e-5</td>
<td>4.3e-4</td>
</tr>
</tbody>
</table>

Table 12: Simultaneously finding of all roots of \( f_7, s \).

\[
f_7(s) = \sin^3 ((s - 1)/2)\sin^3 ((s - 2)/2)\sin^3 ((s - 2.5)/2)
\]
\[
\zeta_1 = 1, \quad \zeta_2 = 2, \quad \zeta_3 = 2.5, \quad \text{each with multiplicity 3}
\]
\[
S_1^{(0)} = -0.2, \quad S_2^{(0)} = 1.7, \quad S_3^{(0)} = 3
\]

<table>
<thead>
<tr>
<th>Method</th>
<th>CO</th>
<th>CUP</th>
<th>( n )</th>
<th>( e1 )</th>
<th>( e2 )</th>
<th>( e3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>PJ6</td>
<td>6</td>
<td>0.047</td>
<td>3</td>
<td>5.5e-16</td>
<td>2.7e-8</td>
<td>2.2e-17</td>
</tr>
<tr>
<td>SM1–SM3</td>
<td>6</td>
<td>0.031</td>
<td>3</td>
<td>1.8e-21</td>
<td>1.7e-13</td>
<td>3.9e-18</td>
</tr>
</tbody>
</table>

\[
S_1^{(0)} = 3.5 + 0.3i, \quad S_2^{(0)} = 3.5 - 0.3i, \quad S_3^{(0)} = -0.3 + 0.01i, \quad S_4^{(0)} = 1.8 + 0.01i.
\]

Example 3. (see [33], for simultaneous determination of distinct and multiple roots).

Here, we consider another standard test function for the demonstration of convergence behavior of newly constructed methods.

Consider

\[
f_{7, s} = \sin \left( \frac{s - 1}{2} \right) \sin \left( \frac{s - 2}{2} \right) \sin \left( \frac{s - 2.5}{2} \right).
\]
with exact roots $\zeta_1 = 1$, $\zeta_2 = 2$, and $\zeta_3 = 2.5$ as shown in Figure 8. The initial guessed values have been taken as

\begin{align}
(0) \quad s_1 & = -0.2, \\
(0) \quad s_2 & = 1.7, \\
(0) \quad s_3 & = 3.
\end{align}

(53)

For distinct roots, we use method (34), and for multiple roots, method (37):

\begin{equation}
\begin{aligned}
f_{7-2}(s) = \sin^3 \left( \frac{s - 1}{2} \right) \sin^3 \left( \frac{s - 2}{2} \right) \sin^3 \left( \frac{s - 2.5}{2} \right). \\
\end{aligned}
\end{equation}

(54)

Figures 9(e)–9(g) show the residual fall of iterative methods (Q1–Q3 and E1–E9), Figures 9(a), 9(b), and 9(d)
Figure 8: Exact real roots of nonlinear equation $f_7(s)$.

Figure 9: Continued.
present the residual fall of simultaneous iterative methods (SM1–SM3 and PJ6) for nonlinear functions \( f_5(s) \), \( f_6(s) \), and \( f_{7-1}(s) \) when the multiplicity of roots is simple, and Figure 9(c) presents the residual fall for multiple roots for a nonlinear function \( f_{7-2}(s) \), respectively.

Table 13 shows the values of parameters \( \alpha \) and \( \beta \) used in iterative methods Q1–Q3 for test Examples 1–3.

7. Conclusion

We have developed here three families of single-step single root finding methods of optimal convergence order four and three families of simultaneous methods of order six, respectively. From Tables 7–13 and Figures 1–5 and 8, we observe that our methods (Q1–Q3 and SM1–SM3) are superior in terms of efficiency, stability, CPU time, and residual error as compared to the methods E1–E9 and PJ6 method, respectively.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Authors’ Contributions

All authors contributed equally in the preparation of this manuscript.

Acknowledgments

This research work was supported by all the authors of this manuscript.
References


