OPEN ACCESS
Fractals (2026) 2640014 (15 pages)

DOI: 10.1142/S0218348X26400141

World Scientific
www.worldscientific.com

ON TWO-STEP PARALLEL COMPUTER ALGORITHM FOR ALL NONLINEAR EQUATIONS ROOTS WITH ENGINEERING APPLICATIONS

MUDASSIR SHAMS
Department of Mathematics, Faculty of Arts and Science
Balikesir University, 10145 Balıkesir, Turkey

Department of Mathematics and Statistics, Riphah
International University, Islamabad 44000, Pakistan

NASREEN KAUSAR
Department of Mathematics, Faculty of Arts and Science
Balikesir University, 10145 Balıkesir, Turkey

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

PRAVEEN AGARWAL
Department of Mathematics, Anand International
College of Engineering, Jaipur, India

Nonlinear Dynamics Research Center (NDRC)
Ajman University, Ajman, UAE

NOUF ABDULRAHMAN ALQAHTANI*,‡ and MDI BEGUM JEELANI†,‡
Department of Mathematics and Statistics
College of Science Imam Mohammad Ibn Saud Islamic
University (IMSIU), Riyadh, Saudi
**naswedan@imamu.edu.sa*
†mbshaikh@imamu.edu.sa

---

‡Corresponding authors.

Received May 27, 2024
Accepted August 25, 2025
Published March 20, 2026

Abstract

In this paper, we develop an optimal family of two-step single root finding methods with order 4. Furthermore, we modify these numerical schemes to locate all nonlinear real and complex roots at the same time. The order of convergence is demonstrated using convergence analysis to be four for a single root-finding technique and for families of parallel computer methods. Numerical test examples reveal that the developed parallel techniques achieve superior stability and consistency compared to existing approaches.

Keywords: Polynomial; Simultaneous Methods; Computational Time; Fractal; Convergence; Percentage Computational Cost.

1. INTRODUCTION

Consider the nonlinear equation

$$\Game(\varsigma) = 0, \tag{1}$$

whose solutions are the n distinct roots $\xi_1, \ldots, \xi_n$. The solution of (1) is foundational to scientific and engineering disciplines, offering a powerful framework for modeling and understanding complex phenomena that cannot be adequately described by linear systems. Their importance spans across various fields, from physics and chemistry to biology, engineering, and beyond. In science, nonlinear equations are essential for modeling natural phenomena such as chaotic systems, wave propagation, and pattern formation. They underpin our understanding of complex systems like weather patterns,[1] ecological dynamics,[2] and brain function,[3] where nonlinear interactions drive emergent behavior and complex dynamics. In engineering, nonlinear equations are crucial for designing and analyzing systems ranging from electrical circuits and mechanical structures to control systems and communication networks. They allow engineers to account for nonlinear effects such as saturation, hysteresis, and feedback, ensuring the accurate prediction of system behavior and enabling the development of robust and efficient technologies.[4] Nonlinear equations also play a vital role in optimization, enabling the solution of complex problems in areas such as signal processing, machine learning, and logistics. Furthermore, they have applications in quantum physics, where they are used to describe phenomena such as quantum interference, nonlinear optics, and the behavior of quantum systems under extreme conditions. Overall, nonlinear equations are essential to scientific research and technological advancement because they offer a potent instrument for delving into the intricacies of the natural world and creating solutions for the most difficult issues that confront humanity. In general, the numerical iterative techniques used to estimate the roots of (1) can be divided into two primary categories: Iterative algorithms that calculated one root at a time, see e.g. Refs. 5–15 and the iterative techniques which find all roots of (1) simultaneously. For particulars on simultaneous techniques, their convergence order, computational cost, C-time and parallel execution of the iterative algorithm on computer algebra system can be seen in the work of Aberth,[16] Consnard and Fraigniaud,[17] Proinov,[18] Mir *et al.*,[19] Kanno *et al.*,[20] Nourein,[21] Proinov and Vasileva,[22] Nedzhibov,[23] Cholakov[24] and references cited therein. Despite their advancements, classical parallel schemes still suffer from the following drawbacks.

- Classical parallel approaches exhibit low convergence rates when applied to the simultaneous computation of all roots of nonlinear equations.
- Higher-order extensions are often unstable or require extra function evaluations, reducing efficiency.
- Existing classical schemes can fail to converge or require high precision.
- On distributed-memory systems, this results in high latency and limits strong scalability.
- Root approximations in classical parallel scheme may converge at different rates.

Unlike existing methods, this study aims to overcome these limitations by developing efficient simultaneous iterative algorithms, with particular emphasis on analyzing their global convergence behavior when initial values are selected far from the exact solution.

The remainder of the paper is organized as follows:

- — **Section 2** — Development and analysis of new parallel schemes for simultaneous root computation.
- — **Section 3** — Discussion of computational efficiency.
- — **Section 4** — Validation of consistency and efficiency against existing methods using biomedical engineering applications.
- — **Section 5** — Concluding remarks.

2. CONSTRUCTION AND CONVERGENCE FRAMEWORK OF NUMERICAL SCHEME

Newton's method is a classical approach for finding single roots of (1):

$$\varsigma^{[k+1]} = \varsigma^{[k]} - \frac{\backepsilon(\varsigma^{[k]})}{\backepsilon'(\varsigma^{[k]})}, \quad k = 1, 2, \ldots \tag{2}$$

which has second-order convergence with efficiency index 1.34. Using Weierstrass' correction[22]:

$$w(\varsigma_i^{[k]}) = \frac{\backepsilon(\varsigma_i^{[k]})}{\prod_{j=1, j\neq i}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})}, \tag{3}$$

we obtain the Weierstrass method for all roots of (1):

$$\varsigma_i^{[k+1]} = \varsigma_i^{[k]} - \frac{\backepsilon(\varsigma_i^{[k]})}{\prod_{j=1, j\neq i}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})}. \tag{4}$$

This method is second-order convergent. Ehrlich's third-order scheme[16] reads

$$\varsigma_i^{[k+1]} = \varsigma_i^{[k]} - \frac{1}{\frac{1}{N(\varsigma_i^{[k]})} - \sum_{j=1, j\neq i}^{n} \frac{1}{\varsigma_i^{[k]} - \varsigma_j^{[k]}}}, \tag{5}$$

where

$$N(\varsigma_i^{[k]}) = \frac{\backepsilon(\varsigma_i^{[k]})}{\backepsilon'(\varsigma_i^{[k]})}$$

denotes the Newton correction at the kth iteration. We propose the following iterative method:

$$\sigma^{[k]} = \varsigma^{[k]} - \frac{\backepsilon(\varsigma^{[k]})}{\backepsilon'(\varsigma^{[k]})},$$

$$\varsigma^{[k+1]} = \sigma^{[k]} - \frac{\backepsilon(\varsigma^{[k]})}{\backepsilon'(\varsigma^{[k]})}[\alpha^{[1]}((\backepsilon(\sigma^{[k]}))^2 + (\acute{L}(u^{[k]}))^2)], \tag{6}$$

where

$$u^{[k]} = \frac{\backepsilon(\sigma^{[k]})}{\backepsilon(\varsigma^{[k]})}, \quad \alpha^{[1]} \in \mathbb{R}, \quad \text{and } \acute{\mathrm{L}}(u^{[k]})$$

is a weight function satisfying certain assumptions as required by the convergence theorem of the scheme.

Theorem 1. *Let* $\xi \in \mathfrak{I}$ *be a root of a sufficiently differentiable function* $\backepsilon : \mathfrak{I} \subseteq \mathbb{R} \to \mathbb{R}$. *If* ς_0 *is sufficiently close to* ξ *and* $\acute{\mathrm{L}}$ *satisfies* $\acute{\mathrm{L}}(0) = 1, \acute{\mathrm{L}}'(0) = 1$ *and* $\acute{\mathrm{L}}''(0) < \infty$, *then* (6) *has fourth-order convergence and the following error equation holds:*

$$\mathfrak{e}_{i+1} = \left[4\left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)^3 - \left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)\left(\frac{\backepsilon'''(\xi)}{6\backepsilon'(\xi)}\right) - \left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)^3 \acute{\mathrm{L}}''(0)\right](\mathfrak{e}_i)^4 + O((\mathfrak{e}_i)^3), \tag{7}$$

where $\mathcal{C}_j = \frac{\backepsilon^j(\xi)}{j!\backepsilon'(\xi)}; j \geq 2$.

Proof. Let ξ be a single root of $\backepsilon$ and $\varsigma^{[k]} = \xi + \mathfrak{e}_i$. Expanding $\backepsilon$ around ξ using Taylor series and noting $\backepsilon(\xi) = 0$, we have

$$\backepsilon(\varsigma^{[k]}) = \backepsilon'(\xi)(\mathfrak{e}_i + \mathcal{C}_2\mathfrak{e}_i^2 + \mathcal{C}_3\mathfrak{e}_i^3 + \mathcal{C}_4\mathfrak{e}_i^4) + O(\mathfrak{e}_i^5), \tag{8}$$

$$\backepsilon'(\varsigma^{[k]}) = \backepsilon'(\xi)(1 + 2\mathcal{C}_2\mathfrak{e}_i + 3\mathcal{C}_3\mathfrak{e}_i^2 + 4\mathcal{C}_4\mathfrak{e}_i^3) + O(\mathfrak{e}_i^4). \tag{9}$$

Dividing (8) by (9), we get

$$\frac{\backepsilon(\varsigma^{[k]})}{\backepsilon'(\varsigma^{[k]})} = \mathfrak{e}_i - \mathcal{C}_2\mathfrak{e}_i^2 + (2\mathcal{C}_2 + 2\mathcal{C}_3)\mathfrak{e}_i^3 + \cdots \tag{10}$$

and

$$\begin{aligned} \sigma^{[k]} &= \mathcal{C}_2\mathfrak{e}_i^2 + (-2\mathcal{C}_2^2 + 2\mathcal{C}_3)\mathfrak{e}_i^3 + \cdots, \\ \backepsilon(\sigma^{[k]}) &= \mathcal{C}_2\mathfrak{e}_i^2 + (-4\mathcal{C}_2^2 + 2\mathcal{C}_3)\mathfrak{e}_i^3 + \cdots. \end{aligned} \tag{11}$$

Let

$$u = \frac{\backepsilon(\sigma^{[k]})}{\backepsilon(\varsigma^{[k]})} = \mathcal{C}_2\mathfrak{e}_i + (-3\mathcal{C}_2^2 + 2\mathcal{C}_3)\mathfrak{e}_i^2 + \cdots \tag{12}$$

and expand $\acute{\mathrm{L}}(u)$ about the origin:

$$\acute{\mathrm{L}}(u) = \acute{\mathrm{L}}(0) + \acute{\mathrm{L}}'(0)u + \frac{1}{2}\acute{\mathrm{L}}''(0)u^2 + \cdots. \tag{13}$$

Finally, the error for the next iteration is

$$\mathfrak{e}_{i+1} = \sigma^{[k]} - \frac{\backepsilon(x^{[k]})}{\backepsilon'(x^{[k]})}[\alpha(\backepsilon(\mathrm{y}^{[k]}))^3 + (\acute{\text{L}}(u))^3]$$

$$= [(-3\acute{\text{L}}^3(0)\mathcal{C}_2 + \mathcal{C}_2)\mathfrak{e}_i^2 + \cdots]. \tag{14}$$

Setting $\acute{\text{L}}(0) = \acute{\text{L}}'(0) = 1$, we obtain (15), proving fourth-order convergence:

$$\mathfrak{e}_{i+1} = \left[4\left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)^3 - \left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)\left(\frac{\backepsilon'''(\xi)}{6\backepsilon'(\xi)}\right) - \left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)^3 \acute{\text{L}}''(0)\right](\mathfrak{e}_i)^4 + O((\mathfrak{e}_i)^5). \tag{15}$$

Hence proves fourth-order convergence. □

2.1. The Concrete Fourth-Order Methods

2.1.1. *The concrete fourth-order methods*

Using $\acute{\text{L}}(0) = 1, \acute{\text{L}}'(0) = 1$ and $\acute{\text{L}}''(0) < \infty$, we construct some concrete methods of (6) as:

Method 1 (abbreviated as SM$^{\backepsilon^{[1]}}$): If we choose

$$\acute{\text{L}}(u^{[k]}) = 1 + \frac{u^{[k]}}{1 + \alpha^{[2]}u^{[k]}}$$

so that $\acute{\text{L}}(0) = 1, \acute{\text{L}}'(0) = 1$, then we have

$$\sigma^{[k]} = \varsigma^{[k]} - \frac{\backepsilon(\varsigma^{[k]})}{\backepsilon'(\varsigma^{[k]})},$$

$$\varsigma^{[k+1]} = \sigma^{[k]} - \frac{\backepsilon(\sigma^{[k]})}{\backepsilon'(\varsigma^{[k]})}\left[\alpha^{[1]}\left((\backepsilon(\sigma^{[k]}))^2 + \left(1 + \frac{\frac{\backepsilon(\sigma^{[k]})}{\backepsilon(\varsigma^{[k]})}}{1 + \alpha^{[2]}\frac{\backepsilon(\sigma^{[k]})}{\backepsilon(\varsigma^{[k]})}}\right)^2\right)\right]. \tag{16}$$

The numerical scheme satisfies the following error equation:

$$\mathfrak{e}_{i+1} = \left[6\left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)^3 - \left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)\left(\frac{\backepsilon'''(\xi)}{6\backepsilon'(\xi)}\right)\right] \times (\mathfrak{e}_i)^4 + O((\mathfrak{e}_i)^5). \tag{17}$$

Method 2 (abbreviated as SM$^{\backepsilon^{[2]}}$): If we choose

$$\acute{\text{L}}(u^{[k]}) = 1 + \frac{u^{[k]}}{1 + u^{[k]} + \alpha^{[2]}(u^{[k]})^2}$$

so that $\acute{\text{L}}(0) = 1, \acute{\text{L}}'(0) = \frac{2}{3}$, then we have

$$\sigma^{[k]} = \varsigma^{[k]} - \frac{\backepsilon(\varsigma^{[k]})}{\backepsilon'(\varsigma^{[k]})},$$

$$\varsigma^{[k+1]} = \sigma^{[k]} - \frac{\backepsilon(\varsigma^{[k]})}{\backepsilon'(\varsigma^{[k]})}\left[\alpha^{[1]}\left((\backepsilon(\sigma^{[k]}))^2 + \left(1 + \frac{\frac{\backepsilon(\sigma^{[k]})}{\backepsilon(\varsigma^{[k]})}}{1 + \frac{\backepsilon(\sigma^{[k]})}{\backepsilon(\varsigma^{[k]})} + \alpha^{[2]}(\frac{\backepsilon(\sigma^{[k]})}{\backepsilon(\varsigma^{[k]})})^2}\right)^2\right)\right]. \tag{18}$$

The numerical scheme satisfies the following error equation:

$$\mathfrak{e}_{i+1} = \left[2\alpha^{[1]}\left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)^3 + 4\left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)^3 - \left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)\left(\frac{\backepsilon'''(\xi)}{6\backepsilon'(\xi)}\right)\right](\mathfrak{e}_i)^4 + O((\mathfrak{e}_i)^5). \tag{19}$$

Method 3 (abbreviated as SM$^{\backepsilon^{[3]}}$): If we choose

$$\acute{\text{L}}(u^{[k]}) = u^{[k]} + \frac{1 + \alpha^{[2]}(u^{[k]})^2}{1 - \alpha^{[2]}(u^{[k]})^2}$$

so that $\acute{\text{L}}(0) = 1, \acute{\text{L}}'(0) = \frac{2}{3}$, then we have

$$\sigma^{[k]} = \varsigma^{[k]} - \frac{\backepsilon(\varsigma^{[k]})}{\backepsilon'(\varsigma^{[k]})},$$

$$\varsigma^{[k+1]} = \sigma^{[k]} - \frac{\backepsilon(\varsigma^{[k]})}{\backepsilon'(\varsigma^{[k]})}\left[\alpha^{[1]}\left((\backepsilon(\sigma^{[k]}))^2 + \left(\frac{\backepsilon(\sigma^{[k]})}{\backepsilon(\varsigma^{[k]})} + \frac{1 + \alpha^{[2]}\left(\frac{\backepsilon(\sigma^{[k]})}{\backepsilon(\varsigma^{[k]})}\right)^2}{1 - \alpha^{[2]}\left(\frac{\backepsilon(\sigma^{[k]})}{\backepsilon(\varsigma^{[k]})}\right)^2}\right)^2\right)\right]. \tag{20}$$

The numerical scheme satisfies the following error equation:

$$\mathfrak{e}_{i+1} = \left[-4\alpha^{[1]}\left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)^3 + 2\left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)^3 - \left(\frac{\backepsilon''(\xi)}{2\backepsilon'(\xi)}\right)\left(\frac{\backepsilon'''(\xi)}{6\backepsilon'(\xi)}\right)\right](\mathfrak{e}_i)^4 + O((\mathfrak{e}_i)^5). \tag{21}$$

Generalization to Simultaneous Methods

Using (3), we convert (6) into a simultaneous iterative method for approximating all nonlinear equation roots as follows:

Simultaneous Method 1 (SSM$^{\eth^{[1]}}$): Using the concrete method SM$^{\eth^{[1]}}$, the iterative scheme is given by

$$\sigma_i^{[k]} = \varsigma_i^{[k]} - \frac{\eth(\varsigma_i^{[k]})}{\prod_{j=1,j\neq i}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})},$$

$$\varsigma_i^{[k+1]} = \sigma_i^{[k]} - \frac{\eth(\sigma_i^{[k]})}{\prod\limits_{j=1,j\neq i}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})}[\alpha^{[1]}(\eth(\sigma_i^{[k]}))^2 + (1+A)^2], \tag{22}$$

where

$$A = \frac{\frac{\eth(\sigma_i^{[k]})}{\eth(\varsigma_i^{[k]})}}{1+\alpha^{[2]}\frac{\eth(\sigma_i^{[k]})}{\eth(\varsigma_i^{[k]})}}.$$

Simultaneous Method 2 (SSM$^{\eth^{[2]}}$): By choosing the concrete method SM$^{\eth^{[2]}}$, we have

$$\sigma_i^{[k]} = \varsigma_i^{[k]} - \frac{\eth(\varsigma_i^{[k]})}{\prod_{\substack{j=1\\ j\neq i}}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})}, \tag{23}$$

$$\varsigma_i^{[k+1]} = \sigma_i^{[k]} - \left[\frac{\eth(\sigma_i^{[k]})}{\prod_{\substack{j=1\\ j\neq i}}^{n}(\sigma_i^{[k]} - \sigma_j^{[k]})}\cdot\frac{\prod_{\substack{j=1\\ j\neq i}}^{n}(\sigma_i^{[k]} - \sigma_j^{[k]})}{\prod_{\substack{j=1\\ j\neq i}}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})}\right]\cdot\left[\alpha^{[1]}(\eth(\sigma_i^{[k]}))^2 + \left(1+\frac{\frac{\eth(\sigma_i^{[k]})}{\eth(\varsigma_i^{[k]})}}{1+\frac{\eth(\sigma_i^{[k]})}{\eth(\varsigma_i^{[k]})}+\alpha^{[2]}(\frac{\eth(\sigma_i^{[k]})}{\eth(\varsigma_i^{[k]})})^2}\right)^2\right]. \tag{24}$$

Simultaneous Method 3 (SSM$^{\eth^{[3]}}$): By choosing the concrete method SM$^{\eth^{[3]}}$, we have

$$\sigma_i^{[k]} = \varsigma_i^{[k]} - \frac{\eth(\varsigma_i^{[k]})}{\prod_{\substack{j=1\\ j\neq i}}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})}, \tag{25}$$

$$\varsigma_i^{[k+1]} = \sigma_i^{[k]} - \left[\frac{\eth(\sigma_i^{[k]})}{\prod_{\substack{j=1\\ j\neq i}}^{n}(\sigma_i^{[k]} - \sigma_j^{[k]})}\cdot\frac{\prod_{\substack{j=1\\ j\neq i}}^{n}(\sigma_i^{[k]} - \sigma_j^{[k]})}{\prod_{\substack{j=1\\ j\neq i}}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})}\right]\cdot\left[\alpha^{[1]}(\eth(\sigma_i^{[k]}))^2 + \left(1+\frac{\frac{\eth(\sigma_i^{[k]})}{\eth(\varsigma_i^{[k]})}}{1+\frac{\eth(\sigma_i^{[k]})}{\eth(\varsigma_i^{[k]})}+\alpha^{[2]}\left(\frac{\eth(\sigma_i^{[k]})}{\eth(\varsigma_i^{[k]})}\right)^2}\right)^2\right]. \tag{26}$$

In the following theorem, we prove the convergence order of SSM$^{\eth^{[1]}}$–SSM$^{\eth^{[3]}}$. Thus, we construct three new simultaneous iterative methods (22)–(26).

Convergence Analysis

Here, we prove convergence order of simultaneous methods SSM$^{\eth^{[1]}}$–SSM$^{\eth^{[3]}}$ using following convergence theorem.

Theorem 2. *Let* $\zeta_1, \ldots, \zeta_\sigma$ *be simple zero of nonlinear equation and for sufficiently close initial distinct estimation* $\varsigma_1^{[0]}, \ldots, \varsigma_n^{[0]}$ *of the roots, respectively, then SSM*$^{\eth^{[1]}}$*–SSM*$^{\eth^{[3]}}$ *has convergence order* 4.

Proof. Let $\epsilon_i = \varsigma_i^{[k]} - \zeta_i$, $\epsilon_i' = \sigma_i^{[k]} - \zeta_i$, and $\epsilon_i'' = \varsigma_i^{[k+1]} - \zeta_i$ be the errors in r_i, u_i, and z_i, respectively.

From the first step of SSM$^{\backepsilon[1]}$, we have

$$\sigma_i^{[k]} - \zeta_i = \varsigma_i^{[k]} - \zeta_i - \frac{\backepsilon(\varsigma_i^{[k]})}{\prod_{\substack{j=1\\ j\neq i}}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})}, \tag{27}$$

$$\epsilon_i' = \epsilon_i - \vartheta_i^*(\varsigma_i^{[k]}) = \epsilon_i - \epsilon_i \frac{\vartheta_i^*(\varsigma_i^{[k]})}{\epsilon_i}, \tag{28}$$

$$\epsilon_i' = \epsilon_i(1 - Q_i^{[1]}), \tag{29}$$

where

$$Q_i^{[1]} = \frac{\vartheta_i^*(\varsigma_i^{[k]})}{\epsilon_i} = \prod_{\substack{j=1\\ j\neq i}}^{n} \left[\frac{\varsigma_i^{[k]} - \zeta_j}{\varsigma_i^{[k]} - \varsigma_j^{[k]}}\right], \tag{30}$$

$$\frac{\varsigma_i^{[k]} - \zeta_j}{\varsigma_i^{[k]} - \varsigma_j^{[k]}} = 1 + \frac{\varsigma_j^{[k]} - \zeta_j}{\varsigma_i^{[k]} - \varsigma_j^{[k]}} + O(|\epsilon|^2), \tag{31}$$

$$\begin{aligned} Q_i^{[1]} &= \prod_{\substack{j=1\\ j\neq i}}^{n} \left[\frac{\varsigma_i^{[k]} - \zeta_j}{\varsigma_i^{[k]} - \varsigma_j^{[k]}}\right] \\ &= (1 + O(|\epsilon|))^{n-1} \\ &= 1 + (n-1)O(|\epsilon|) \\ &= 1 + O(|\epsilon|), \end{aligned} \tag{32}$$

$$Q_i^{[1]} - 1 = O(|\epsilon|), \tag{33}$$

thus, we get

$$\epsilon_i' = \epsilon_i\, O(|\epsilon|) = O(|\epsilon|^2). \tag{34}$$

Consider the second-step of parallel scheme as

$$\varsigma_i^{[k+1]} - \zeta_i = \sigma_i^{[k]} - \zeta_i - \left[\frac{\backepsilon(\sigma_i^{[k]})}{\prod_{\substack{j=1\\ j\neq i}}^{n}(\sigma_i^{[k]} - \sigma_j^{[k]})} \cdot \frac{\prod_{\substack{j=1\\ j\neq i}}^{n}(\sigma_i^{[k]} - \sigma_j^{[k]})}{\prod_{\substack{j=1\\ j\neq i}}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})} \cdot [\alpha^{[1]}(\sigma_i^{[k]})^2 + (1 + B_1^{[*]})^2]\right], \tag{35}$$

where $B_1^{[*]} = \frac{\backepsilon(\sigma_i^{[k]})/\backepsilon(\varsigma_i^{[k]})}{1+\alpha^{[2]}(\backepsilon(\sigma_i^{[k]})/\backepsilon(\varsigma_i^{[k]}))+(\backepsilon(\sigma_i^{[k]})/\backepsilon(\varsigma_i^{[k]}))^2}$,

$$\epsilon_i'' = \epsilon_i' - \vartheta_i^*(\sigma_i^{[k]}) \tag{36}$$

$$= \epsilon_i' - \epsilon_i' \frac{\vartheta_i^*(\sigma_i^{[k]})}{\epsilon_i'} \cdot \frac{\prod_{\substack{j=1\\ j\neq i}}^{n}(\sigma_i^{[k]} - \sigma_j^{[k]})}{\prod_{\substack{j=1\\ j\neq i}}^{n}(\varsigma_i^{[k]} - \varsigma_j^{[k]})} \tag{37}$$

$$\cdot \left[\alpha^{[1]}(\sigma_i^{[k]})^2 + \left(1 + \frac{\frac{\backepsilon(\sigma_i^{[k]})}{\backepsilon(\varsigma_i^{[k]})}}{1 + B_2^{[*]}}\right)^2\right]$$

$$= \epsilon_i'(1 - Q_i^{[2]}), \tag{38}$$

where

$$B_2^{[*]} = \alpha^{[2]} \frac{\backepsilon(\sigma_i^{[k]})}{\backepsilon(\varsigma_i^{[k]})} + \left(\frac{\backepsilon(\sigma_i^{[k]})}{\backepsilon(\varsigma_i^{[k]})}\right)^2,$$

$$Q_i^{[2]} = \frac{\vartheta_i^*(\sigma_i^{[k]})}{\epsilon_i'} = \prod_{\substack{j=1\\ j\neq i}}^{n} \left[\frac{\sigma_i^{[k]} - \zeta_j}{\sigma_i^{[k]} - \overset{*}{\sigma}_j^{[k]}}\right], \tag{39}$$

$$\begin{aligned} \frac{\sigma_i^{[k]} - \zeta_j}{\sigma_i^{[k]} - \sigma_j^{[k]}} &= 1 + \frac{\sigma_j^{[k]} - \zeta_j}{\sigma_i^{[k]} - \sigma_j^{[k]}} \\ &= 1 + O(|\epsilon^2|), \end{aligned} \tag{40}$$

$$\begin{aligned} Q_i^{[2]} &= \prod_{\substack{j=1\\ j\neq i}}^{n} \left[\frac{\sigma_i^{[k]} - \zeta_j}{\sigma_i^{[k]} - \sigma_j^{[k]}}\right] \\ &= 1 + O(|\epsilon'|), \end{aligned} \tag{41}$$

$$Q_i^{[2]} - 1 = O(|\epsilon'|). \tag{42}$$

Assume $|\epsilon_i| = |\epsilon_j| = |\epsilon|$, then, we get

$$\begin{aligned} \epsilon_i'' &= \epsilon_i'\, O(|\epsilon'|) \\ &= O(|\epsilon'|^2) \\ &= O(|\epsilon|^4), \end{aligned} \tag{43}$$

$$\begin{aligned} \backepsilon(\varsigma_i^{[k]}) &= (\varsigma_1^{[k]} - \zeta_1) \cdots (\varsigma_i^{[k]} - \zeta_i) \\ &= \epsilon_i \prod_{\substack{j=1\\ j\neq i}}^{n} (\varsigma_i^{[k]} - \zeta_j), \end{aligned} \tag{44}$$

$$\begin{aligned} \backepsilon(\sigma_i^{[k]}) &= (\sigma_1^{[k]} - \zeta_1) \cdots (\sigma_i^{[k]} - \zeta_i), \\ &= \epsilon_i' \prod_{\substack{j=1\\ j\neq i}}^{n} (\sigma_i^{[k]} - \zeta_j), \end{aligned}$$

$$
\begin{aligned}
\frac{\ni'(\sigma_i^{[k]})}{\ni'(z_i^{[k]})} &= \frac{\ni(\sigma_i^{[k]})\ni(z_i^{[k]})}{\ni(\sigma_i^{[k]})\ni(z_i^{[k]})} \cdot \prod_{\substack{j=1\\ j\neq i}}^{n} \left[\frac{\sigma_i^{[k]}-\zeta_j}{z_i^{[k]}-\zeta_j}\right] \\
&= 1, \\
\frac{\ni(\sigma_i^{[k]})}{\ni(\varsigma_i^{[k]})} &= \frac{\epsilon_i'}{\epsilon_i} \prod_{\substack{j=1\\ j\neq i}}^{n} \frac{\sigma_i^{[k]}-\zeta_j}{\varsigma_i^{[k]}-\zeta_j} \\
&= \frac{\epsilon_i(1-Q_i^{[1]})}{\epsilon_i} \prod_{\substack{j=1\\ j\neq i}}^{n} \frac{\varsigma_i^{[k]}-\vartheta_i^*(\varsigma_i^{[k]})-\zeta_j}{\varsigma_i^{[k]}-\zeta_j} \\
&= (1-Q_i^{[1]}) \prod_{\substack{j=1\\ j\neq i}}^{n} \left[1-\frac{\vartheta_i^*(\varsigma_i^{[k]})}{\varsigma_i^{[k]}-\zeta_j}\right] \\
&= (1-Q_i^{[1]}) \prod_{\substack{j=1\\ j\neq i}}^{n} \left[1-\frac{\epsilon_i \prod_{\substack{j=1\\ j\neq i}}^{n} \frac{\varsigma_i^{[k]}-\zeta_j}{\varsigma_i^{[k]}-\varsigma_j^{[k]}}}{\varsigma_i^{[k]}-\zeta_j}\right] \\
&= (1-Q_i^{[1]}) \prod_{\substack{j=1\\ j\neq i}}^{n} \\
&\quad \times \left[1-\epsilon_i \prod_{\substack{j=1\\ j\neq i}}^{n} \frac{\varsigma_i^{[k]}-\zeta_j}{(\varsigma_i^{[k]}-\varsigma_j^{[k]})(\varsigma_i^{[k]}-\zeta_j)}\right],
\end{aligned}
\tag{45}
$$

thus

$$
\begin{aligned}
\frac{\ni(\sigma_i^{[k]})}{\ni(\varsigma_i^{[k]})} &= \frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))^{n-1} \\
&= \frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+(n-1)O(|\epsilon|)) \\
&= \frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|)).
\end{aligned}
\tag{46}
$$

Thus, we get

$$
\begin{aligned}
\epsilon_i'' = \epsilon_i' - \epsilon_i' \frac{\vartheta_i^*(\sigma_i^{[k]})}{\epsilon_i'} \cdot \left[\alpha^{[1]} \left(\epsilon_i' \prod_{\substack{j=1\\ j\neq i}}^{n} (\sigma_i^{[k]}-\zeta_j)\right)^2\right. \\
\left. + \left(1 + \frac{\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))}{1+\alpha^{[2]}\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))}\right)^2\right],
\end{aligned}
\tag{47}
$$

$$
\begin{aligned}
\epsilon_i'' &= \epsilon_i' - \epsilon_i'(1+O(|\epsilon'|)) \\
&\quad \cdot \left[\alpha^{[1]} \left(\epsilon_i' \prod_{\substack{j=1\\ j\neq i}}^{n} (\sigma_i^{[k]}-\zeta_j)\right)^2\right. \\
&\quad + \left(1 + 2\frac{\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))}{1+\alpha^{[2]}\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))}\right. \\
&\quad \left.\left. + \frac{\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))}{1+\alpha^{[2]}\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))} + \cdots\right)\right],
\end{aligned}
\tag{48}
$$

$$
\begin{aligned}
\epsilon_i'' &= \epsilon_i' - \epsilon_i'(1+O(|\epsilon'|)) \\
&\quad \cdot \left[\alpha^{[1]} \left(\epsilon_i' \prod_{\substack{j=1\\ j\neq i}}^{n} (\sigma_i^{[k]}-\zeta_j)\right)^2\right. \\
&\quad + \left(2\frac{\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))}{1+\alpha^{[2]}\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))}\right. \\
&\quad \left.\left. + \frac{\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))}{1+\alpha^{[2]}\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))} + \cdots\right)\right],
\end{aligned}
\tag{49}
$$

$$
\begin{aligned}
\epsilon_i'' = \epsilon_i' - \epsilon_i' + \epsilon_i' O(|\epsilon'|) \cdot \left[1 + 2B_3^{[*]}\right. \\
\left. + B_4^{[*]} + \alpha^{[1]} \left(\epsilon_i' \prod_{\substack{j=1\\ j\neq i}}^{n} (\sigma_i^{[k]}-\zeta_j)\right)^2\right],
\end{aligned}
\tag{50}
$$

where

$$
B_3^{[*]} = \frac{\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))}{1+\alpha^{[2]}\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))},
$$

$$
B_4^{[*]} = \frac{\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))}{1+\alpha^{[2]}\frac{\epsilon_i'}{\epsilon_i}(1-Q_i^{[1]})(1+O(|\epsilon|))},
$$

$$\epsilon_i'' = O(|\epsilon'|^2)\left[1 + 2B_3^{[*]} + B_4^{[*]} + \cdots + \alpha^{[1]} \times \left(\epsilon_i' \prod_{\substack{j=1\\ j\neq i}}^{n} (\sigma_i^{[k]} - \zeta_j)\right)^2\right], \tag{51}$$

$$\epsilon_i'' = O(|\epsilon^2|^2) = O(|\epsilon|)^4. \tag{52}$$

Hence prove the theorem. □

3. COMPUTATIONAL ASPECT

The percentage computational efficiency provides a quantitative measure of how effectively a parallel scheme utilizes computational resources relative to its convergence order. It enables direct comparison between different iterative methods, highlighting which schemes achieve faster convergence with fewer arithmetic operations. This metric is crucial for optimizing performance in high-dimensional problems, ensuring both accuracy and minimal computational cost. We compare the percentage cost of computation of Midrog Petković method^{25} and the new methods $\mathrm{SSM}^{\Game[1]}$–$\mathrm{SSM}^{\Game[3]}$. As presented in Ref. 25, the efficiency index $(\Lambda^{[*]}(m))$ is computed as

$$\Lambda^{[*]}(m) = \frac{\log \mathbf{u}}{\mathbf{D}}, \tag{53}$$

where $\mathbf{D}^{25}$ is computational cost and $\mathbf{u}$ is convergence order of $\mathrm{SSM}^{\Game[1]}$–$\mathrm{SSM}^{\Game[3]}$, respectively. Using (53) and Table 1, the percentage computational cost is

$$\vartheta(\mathrm{SSM}^{\Game[1]} - \mathrm{SSM}^{\Game[3]},\ \mathrm{SSM}^{\Game[*]}) = \frac{\Lambda^{[*]}(\mathrm{SSM}^{\Game[1]} - \mathrm{SSM}^{\Game[3]})}{\Lambda^{[*]}(\mathrm{SSM}^{\Game[*]})} - 1(\%), \tag{54}$$

Table 1 Number of ±, ×, ÷.

Schemes	$\mathbf{SSM}^{\Game[1]}$	$\mathbf{SSM}^{\Game[2]}$	$\mathbf{SSM}^{\Game[3]}$	$\mathbf{SSM}^{\Game[*]}$
±	$6\vartheta^{[**]}$	$7\vartheta^{[**]}$	$9\vartheta^{[**]}$	$8\vartheta^{[**]}$
×	$1\vartheta^{[**]}$	$1\vartheta^{[**]}$	$1\vartheta^{[**]}$	$6\vartheta^{[**]}$
÷	$2\vartheta^{[**]}$	$1\vartheta^{[**]}$	$2\vartheta^{[**]}$	$2\vartheta^{[**]}$

Note: $\vartheta^{[**]} = \mathfrak{m}^2 + \mathrm{O}(\mathfrak{m})$.

where $\mathrm{SSM}^{\Game[*]}$ is Petković method^{25} of order 6. Figures 1a–1f graphically clarify these percentage proficiencies. It is obvious from Figs. 1a–1c that $\mathrm{SSM}^{\Game[1]}$–$\mathrm{SSM}^{\Game[3]}$ are more efficient as compared to $\mathrm{SSM}^{\Game[*]}$.

4. NUMERICAL RESULTS

Using CAS Maple 2022 with 64-digit floating-point arithmetic for all numerical calculations with stopping criteria is as follows:

$$\mathrm{e}_i^{[k]} = \|\varsigma_i^{[k+1]} - \varsigma_i^{[k]}\|_2 < \in = 10^{-30},$$

where $\mathrm{e}_i^{[k]}$ represents the absolute error of norm-2

Numerical tests problems from Refs. 12, 26–30 are provided in Tables 2–4. The implementation is carried out in Maple 2022 on the following computational environment:

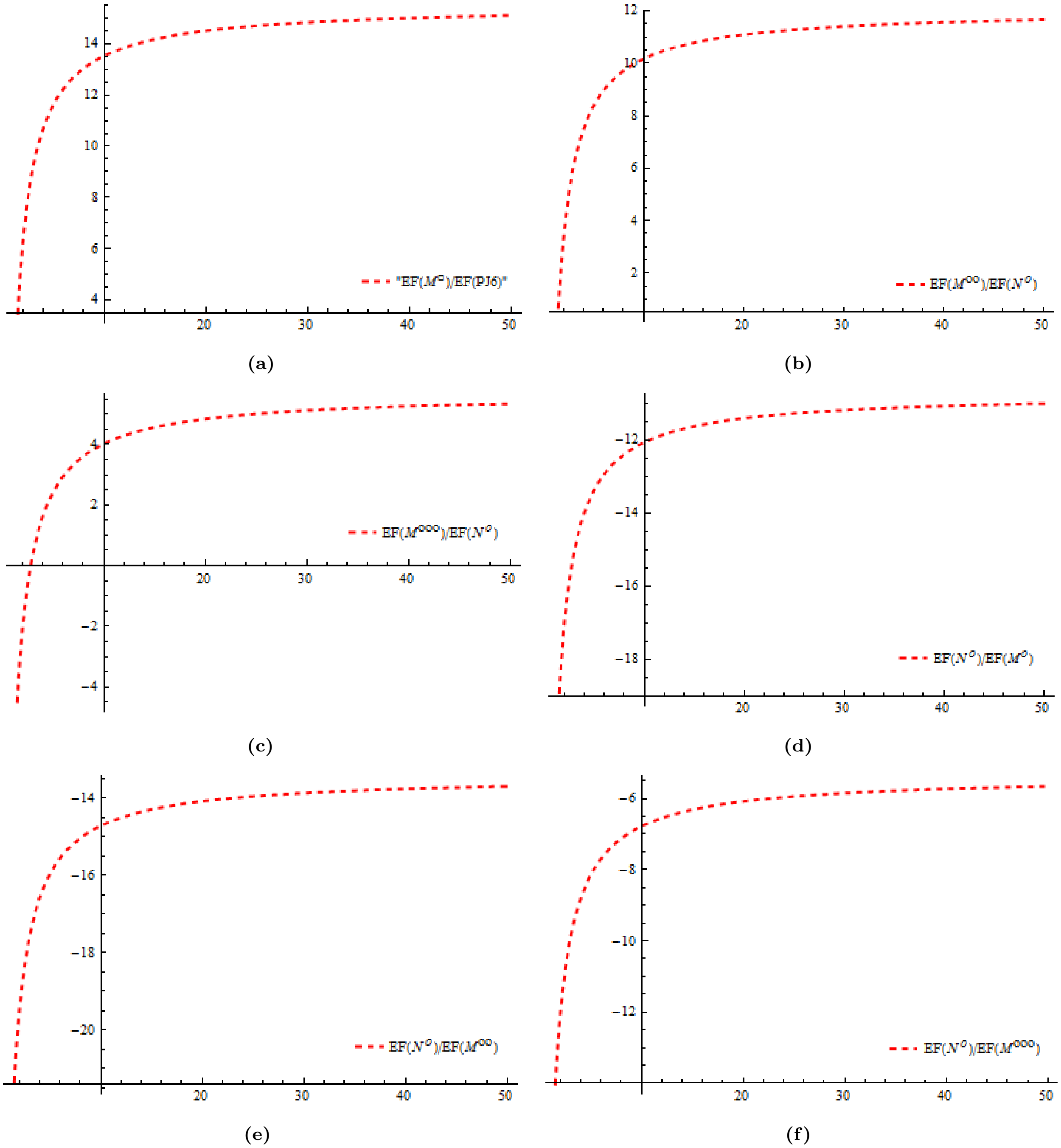
- **Processor:** Intel Core™ i7-8700 CPU @ 3.20 GHz
- **RAM:** 16 GB DDR4
- **Operating System:** Windows 11 Pro, 64-bit
- **Maple Version:** 2022 (64-bit)
- **Toolboxes:** Symbolic Math Toolbox, Parallel Computing Toolbox

Here, we solve some standard nonlinear polynomial having engineering application.

Example 1. The fractional conversion problem in the synthesis of ammonia from nitrogen and hydrogen involves understanding the equilibrium of the reaction and how it relates to the extent of the reaction. The synthesis of ammonia, known as the Haber process, is represented by the following balanced chemical equation:

$$\mathrm{N}_2^{[g]} + 3\mathrm{H}_2^{[g]} \rightleftharpoons 2\mathrm{NH}_3^{[g]}.$$

This equation indicates that one mole of nitrogen gas reacts with three moles of hydrogen gas to produce two moles of ammonia gas. The fractional conversion problem arises from the fact that the reaction does not proceed to completion; instead, it reaches an equilibrium where both reactants and products are present. The degree of conversion, denoted by ς, represents the ratio of the moles of reactant that have been converted to product to the total moles of reactant present initially. At equilibrium, the ratio of the concentrations of the products to the concentrations of the reactants is constant, defined by the equilibrium constant (K). For the



(a) **(b)** **(c)** **(d)** **(e)** **(f)**

Fig. 1 **(a)–(f)** show the percentage computational efficiency of $\text{SSM}^{\partial^{[1]}}$–$\text{SSM}^{\partial^{[3]}}$ and $\text{SSM}^{\partial^{[*]}}$ with respect to each other.

Haber process, the equilibrium constant expression is

$$K = \frac{[\text{NH}_3^{[g]}]^2}{[\text{N}_2^{[g]}][\text{H}_2^{[g]}]^3},$$

where

$[\text{NH}_3^{[g]}]$:= the concentration of ammonia gas,
$[\text{N}_2^{[g]}]$:= the concentration of nitrogen gas,
$[\text{H}_2^{[g]}]$:= the concentration of hydrogen gas.

To solve the fractional conversion problem, you need to find the equilibrium concentrations of each species and then use them to calculate the extent of the reaction (ς) from the following nonlinear equations:

$$\partial(\varsigma) = \frac{8(4-\varsigma)^2\varsigma^2}{(6-3\varsigma)^2(2-\varsigma)} - 0.186. \tag{55}$$

Table 2 Numerical Outcomes for Example 1.

Method	**C-time**	$\mathbf{e}_1^{(4)}$	$\mathbf{e}_2^{(4)}$	$\mathbf{e}_3^{(4)}$	$\mathbf{e}_4^{(4)}$
$\mathrm{SSM}^{\reflectbox{\partial}^{[*]}}$	0.0671	6.9e−22	1.3e−33	0.6e−22	7.7e−22
$\mathrm{SSM}^{\reflectbox{\partial}^{[1]}}$	0.0261	4.48e−45	4.9e−74	0.0	0.0
$\mathrm{SSM}^{\reflectbox{\partial}^{[2]}}$	0.0163	0.98e−55	1.92e−54	0.0	0.0
$\mathrm{SSM}^{\reflectbox{\partial}^{[3]}}$	0.0162	0.89e−45	0.92e−34	0.0	0.0
Residual errors for finding all multiple roots					
$\mathrm{SSM}^{\reflectbox{\partial}^{[*]}}$	0.0232	2.1e−35	6.8e−55	4.1e−33	0.0
$\mathrm{SSM}^{\reflectbox{\partial}^{[1]}}$	0.0233	0.0	0.0	4.1e−33	0.0
$\mathrm{SSM}^{\reflectbox{\partial}^{[2]}}$	0.0313	0.0	0.0	4.1e−33	0.0
$\mathrm{SSM}^{\reflectbox{\partial}^{[3]}}$	0.0323	0.0	0.0	0.0	0.0

The exact roots of the following nonlinear equations are

$$\xi_{1,2} = 3.9 \pm 0.31i, \quad \xi_3 = -0.384094,$$
$$\xi_4 = 0.27775$$

and we choose the following initial guess values to approximate all roots of (55) using scheme $\mathrm{SSM}^{\reflectbox{\partial}^{[1]}}$–$\mathrm{SSM}^{\reflectbox{\partial}^{[3]}}$, $\mathrm{SSM}^{\reflectbox{\partial}^{[*]}}$ as

$$\begin{bmatrix} \varsigma_1^{(0)} = -0.1 + 0.2i, & \varsigma_2^{(0)} = -0.1 - 0.2i, \\ \varsigma_3^{(0)} = 3.5 + 3.2i, & \varsigma_4^{(0)} = 3.9 - 3.2i \end{bmatrix}.$$

The outcomes of the problems using numerical schemes are shown in Table 2. Table 2 clearly presents the dominance convergence rate of the scheme $\mathrm{SSM}^{\reflectbox{\partial}^{[1]}}$–$\mathrm{SSM}^{\reflectbox{\partial}^{[3]}}$ over existing method $\mathrm{SSM}^{\reflectbox{\partial}^{[*]}}$.

Example 2. The solution of eigenvalue problems obtained from matrices of dimensions greater than 5 holds immense importance across numerous disciplines. In fields such as structural engineering, understanding the vibrational modes of complex structures like bridges and buildings is crucial for ensuring their stability and safety. Quantum mechanics relies heavily on eigenvalue solutions to determine energy levels and wave functions of quantum systems, offering insights into the behavior of particles at the atomic and subatomic levels. In summary, the solution of eigenvalue problems from matrices of dimension greater than 5 plays a pivotal role in advancing knowledge, solving complex problems, and driving innovation across a wide range of fields. Here we consider the characteristic equations for the matrix

$$\Lambda = \frac{1}{16} \begin{bmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{bmatrix}.$$

The submatrices are

$$\Lambda_{11} = \begin{bmatrix} -24 & 0 & 0 & 38 & -38 \\ -128 & 0 & 0 & -48 & 48 \\ -32 & 0 & 0 & 8 & -8 \\ 80 & 0 & 0 & -20 & 100 \\ -8 & 0 & 0 & -2 & 82 \end{bmatrix},$$

$$\Lambda_{12} = \begin{bmatrix} 152 & -38 & 36 & 874 \\ 128 & -16 & 64 & 752 \\ 32 & -8 & 16 & 184 \\ 80 & 4 & 40 & 484 \\ 8 & 2 & 4 & 50 \end{bmatrix},$$

$$\Lambda_{21} = \begin{bmatrix} 80 & 0 & 0 & 36 & -36 \\ -168 & 0 & 0 & -58 & 58 \\ 32 & 0 & 0 & -8 & 8 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\Lambda_{22} = \begin{bmatrix} 208 & -36 & 40 & 924 \\ 168 & 42 & 84 & 1002 \\ -32 & 8 & 32 & -184 \\ 0 & 0 & 0 & 48 \end{bmatrix}.$$

The corresponding characteristic polynomial is

$$\begin{aligned} \reflectbox{\partial}(\varsigma) = {} & \varsigma^9 - 29\varsigma^8 + 349\varsigma^7 - 2261\varsigma^6 \\ & + 8455\varsigma^5 - 17663\varsigma^4 + 15927\varsigma^3 + 6993\varsigma^2 \\ & - 24732\varsigma + 12960. \end{aligned} \tag{56}$$

Using a linear algebra approach to find the roots of such higher order characteristic equations is complicated. Consequently, using numerical approaches is one of the best alternatives. The exact root of the

Table 3 Numerical Outcomes for Example 2.

Method	**C-time**	$\mathbf{e}_1^{(4)}$	$\mathbf{e}_2^{(4)}$	$\mathbf{e}_3^{(4)}$	$\mathbf{e}_4^{(4)}$	$\mathbf{e}_5^{(4)}$	$\mathbf{e}_6^{(4)}$	$\mathbf{e}_7^{(4)}$	$\mathbf{e}_8^{(4)}$	$\mathbf{e}_9^{(4)}$
$\mathrm{SSM}^{\reflectbox{\partial}^{[*]}}$	0.057	6.2e−22	1.3e−33	0.6e−22	7.7e−22	2.9e−22	6.9e−22	1.3e−33	0.6e−22	7.7e−22
$\mathrm{SSM}^{\reflectbox{\partial}^{[1]}}$	0.016	4.8e−55	7.9e−64	0.0	0.0	2.8e−55	0.0	4.9e−24	0.0	0.0
$\mathrm{SSM}^{\reflectbox{\partial}^{[2]}}$	0.016	2.2e−55	7.9e−64	0.0	0.0	2.6e−55	0.0	2.9e−74	0.0	0.0
$\mathrm{SSM}^{\reflectbox{\partial}^{[3]}}$	0.016	6.8e−55	7.9e−64	0.0	0.0	2.5e−55	0.0	7.4e−44	0.0	0.0
Residual errors for finding all multiple roots										
$\mathrm{SSM}^{\reflectbox{\partial}^{[*]}}$	0.033	2.1e−35	5.8e−35	4.1e−23	0.0	6.9e−42	6.9e−62	1.3e−33	0.6e−22	7.7e−22
$\mathrm{SSM}^{\reflectbox{\partial}^{[1]}}$	0.033	2.1e−35	5.58e−71	4.1e−43	0.0	4.8e−65	4.8e−65	7.9e−64	0.0	0.0
$\mathrm{SSM}^{\reflectbox{\partial}^{[2]}}$	0.033	2.1e−35	5.8e−41	4.1e−43	0.0	4.8e−65	4.8e−65	7.9e−64	0.0	0.0
$\mathrm{SSM}^{\reflectbox{\partial}^{[3]}}$	0.033	7.3e−84	5.8e−53	0.0	0.0	4.8e−75	4.8e−65	7.9e−64	0.0	0.0

following nonlinear equations is

$$\begin{bmatrix} \xi_1 = 1, \quad \xi_2 = 4, \quad \xi_3 = 5, \\ \xi_4 = 8, \quad \xi_5 = -1, \\ \xi_{6,7,8,9} = 3.0 \end{bmatrix}$$

and we choose the following initial guess values to approximate all roots of (56) using scheme $\mathrm{SSM}^{\reflectbox{\partial}^{[1]}}$–$\mathrm{SSM}^{\reflectbox{\partial}^{[3]}}$,$\mathrm{SSM}^{\reflectbox{\partial}^{[*]}}$ as

$$\begin{bmatrix} \overset{(0)}{\varsigma_1} = -0.1, \quad \overset{(0)}{\varsigma_2} = 3.1, \\ \overset{(0)}{\varsigma}_3 = 5.2, \\ \overset{(0)}{\varsigma}_4 = 7.9, \\ \overset{(0)}{\varsigma_5} = -0.1, \\ \overset{(0)}{\varsigma_6} = 3.1, \\ \overset{(0)}{\varsigma}_7 = 6.7, \\ \overset{(0)}{\varsigma}_8 = 1.5, \\ \overset{(0)}{\varsigma}_9 = 2.9 \end{bmatrix}.$$

The outcomes of the problems using numerical schemes are shown in Table 3. Table 3 clearly presents the dominance convergence rate of the scheme $\mathrm{SSM}^{\reflectbox{\partial}^{[1]}}$–$\mathrm{SSM}^{\reflectbox{\partial}^{[3]}}$ over existing method $\mathrm{SSM}^{\reflectbox{\partial}^{[*]}}$.

Example 3. The Lenard–Jones potential represents one of the most fundamental models in molecular dynamics and computational chemistry, playing a significant role in understanding inter-molecular interactions and the behavior of condensed matter systems. The potential describes the interaction energy between a pair of neutral atoms or molecules as a function of their separation distance. It consists of two terms: a repulsive term that accounts for the Pauli exclusion principle and prevents the atoms from overlapping, and an attractive term arising from dispersion forces.

The importance of solving problems related to the Lenard–Jones potential lies in its widespread applicability across various scientific domains. In physics, it is used to model the behavior of gases, liquids, and solids, providing insights into phase transitions, viscosity, and thermal conductivity. In chemistry, the potential is crucial for understanding molecular bonding, crystal structures, and reaction kinetics. Additionally, in material science, it helps in predicting properties such as elasticity, strength, and adhesion of materials. The problem investigates the interaction in atomic physics and physical chemistry between two neutral atoms or molecules, which is described as

$$\vartheta(\varsigma) = \Theta_0 \left(\left(\frac{\sigma}{\varsigma} \right)^{12} - \left(\frac{\sigma}{\varsigma} \right)^{6} \right),$$

where σ is the place where the inter-particle interaction zeroes out, ς is the separation between the two particles and Θ_0 is the depth of the potential. We compute the shortest distance between the molecules as $\vartheta'(\varsigma) = 0 = \reflectbox{\partial}(\varsigma)$, where

$$\reflectbox{\partial}(\varsigma) = -12 \left(\frac{\sigma^{12}}{\varsigma^{13}} \right) + 6 \left(\frac{\sigma^{6}}{\varsigma^{7}} \right).$$

Table 4 Numerical Outcomes for Example 3.

Method	C-time	$e_1^{(4)}$	$e_2^{(4)}$	$e_3^{(4)}$	$e_4^{(4)}$	$e_5^{(4)}$	$e_6^{(4)}$
$\mathrm{SSM}^{\Gamma^{[*]}}$	0.0572	$6.39e{-}22$	$3.32e{-}33$	$3.64e{-}22$	$3.73e{-}22$	$2.7e{-}22$	$5.75e{-}22$
$\mathrm{SSM}^{\Gamma^{[1]}}$	0.0146	$3.38e{-}35$	$0.92e{-}64$	0.0	0.0	0.0	0.0
$\mathrm{SSM}^{\Gamma^{[2]}}$	0.0124	$3.48e{-}55$	$3.94e{-}44$	0.0	0.0	0.0	0.0
$\mathrm{SSM}^{\Gamma^{[3]}}$	0.0162	$2.85e{-}55$	$4.94e{-}64$	0.0	0.0	0.0	0.0

By choosing $\sigma = 2$, the exact root up to four decimal places is given as

$$\begin{bmatrix} \xi_1 = 2 * 2^{\frac{1}{6}}, \\ \xi_2 = 2\left(\frac{1}{2} + \frac{1}{2}\sqrt{3}i\right)2^{\frac{1}{6}}, \\ \xi_3 = 2\left(-\frac{1}{2} + \frac{1}{2}\sqrt{3}i\right)2^{\frac{1}{6}}, \\ \xi_4 = 2\left(-\frac{1}{2} - \frac{1}{2}\sqrt{3}i\right)2^{\frac{1}{6}}, \\ \xi_3 = 2\left(\frac{1}{2} - \frac{1}{2}\sqrt{3}i\right)2^{\frac{1}{6}}, \\ \xi_4 = -2 * 2^{\frac{1}{6}} \end{bmatrix}$$

and we choose the following initial guess values to approximate all roots of (56) using scheme $\mathrm{SSM}^{\Game^{[1]}}$–$\mathrm{SSM}^{\Game^{[3]}}$, $\mathrm{SSM}^{\Game^{[*]}}$ as

$$\begin{bmatrix} \overset{(0)}{\varsigma}_1 = -0.1 - 0.1i, \\ \overset{(0)}{\varsigma}_2 = 3.1 + 0.2i, \\ \overset{(0)}{\varsigma}_3 = 5.2, \\ \overset{(0)}{\varsigma}_4 = 7.9 - 3.1i, \\ \overset{(0)}{\varsigma}_5 = -0.1 + 0.3i, \quad \overset{(0)}{\varsigma}_6 = 3.1 \end{bmatrix}.$$

The outcomes of the problems using numerical schemes are shown in Table 4. Table 4 clearly presents the dominance convergence rate of the scheme $\mathrm{SSM}^{\Game^{[1]}}$–$\mathrm{SSM}^{\Game^{[3]}}$ over existing method $\mathrm{SSM}^{\Game^{[*]}}$.

Example 4. Consider

$$\Game(\varsigma) = \begin{bmatrix} \begin{pmatrix} (\varsigma+1)^3 \\ (\varsigma+2)^2 \end{pmatrix} \\ \begin{pmatrix} \varsigma - 1.0 \\ -1.0 * i \end{pmatrix}^3 \\ \begin{pmatrix} \varsigma - \\ 1.0 + i \end{pmatrix}^3 \end{bmatrix}$$

with exact real and complex roots:

$$\begin{bmatrix} \xi_1 = -1, \quad \xi_2 = -2, \\ \xi_3 = 1.0 + i, \\ \xi_4 = 1.0 - 1.0i \end{bmatrix}$$

having the following multiplicity:

$$\begin{bmatrix} \sigma_1 = 3, \quad \sigma_2 = 2, \\ \sigma_3 = 3, \quad \sigma_4 = 3 \end{bmatrix}$$

and we choose the following initial guess values to approximate all roots of (56) using scheme

Table 5 Numerical Outcomes for Example 4.

Method	C-time	$\mathbf{e}_1^{(4)}$	$\mathbf{e}_2^{(4)}$	$\mathbf{e}_3^{(4)}$	$\mathbf{e}_4^{(4)}$
$\mathrm{SSM}^{\Game^{[*]}}$	0.057	6.9e−22	1.3e−33	0.6e−22	0.0
$\mathrm{SSM}^{\Game^{[1]}}$	0.016	2.0e−55	0.0	0.0	0.0
$\mathrm{SSM}^{\Game^{[2]}}$	0.016	4.2e−65	0.0	0.0	0.0
$\mathrm{SSM}^{\Game^{[3]}}$	0.016	5.5e−63	0.0	0.0	0.0
Residual errors for finding all multiple roots					
$\mathrm{SSM}^{\Game^{[*]}}$	0.033	2.1e−35	6.8e−55	4.51e−33	0.0
$\mathrm{SSM}^{\Game^{[1]}}$	0.03	1.41e−35	0.0	0.31e−33	0.0
$\mathrm{SSM}^{\Game^{[2]}}$	0.01	1.51e−35	0.0	1.15e−33	0.0
$\mathrm{SSM}^{\Game^{[3]}}$	0.02	7.3e−84	0.0	0.0	0.0

Table 6 Numerical Outcomes for Example 5.

Method	CPU	$\mathbf{e}_1^{(4)}$	$\mathbf{e}_2^{(4)}$	$\mathbf{e}_3^{(4)}$
$\mathrm{SSM}^{\Game^{[*]}}$	0.031	3.5e−21	5.5e−20	2.2e−20
$\mathrm{SSM}^{\Game^{[1]}}$	0.031	3.5e−21	5.5e−20	2.2e−20
$\mathrm{SSM}^{\Game^{[2]}}$	0.031	3.5e−21	5.5e−20	2.2e−20
$\mathrm{SSM}^{\Game^{[3]}}$	0.015	0.0	0.0	0.0

Table 7 Numerical Outcomes for Example 6.

Method	C-time	$e_1^{(4)}$	$e_2^{(4)}$	$e_3^{(4)}$	$e_4^{(4)}$	$e_5^{(4)}$	$e_6^{(4)}$
$\text{SSM}^{\Gamma^{[*]}}$	0.0573	6.9*e*-22	1.3*e*-33	0.6*e*-22	7.7*e*-22	—	—
$\text{SSM}^{\Gamma^{[1]}}$	0.0162	0.0	7.9*e*-64	0.0	0.0	0.0	0.0
$\text{SSM}^{\Gamma^{[2]}}$	0.0132	0.0	7.9*e*-64	0.0	0.0	0.0	0.0
$\text{SSM}^{\Gamma^{[3]}}$	0.0143	0.0	7.9*e*-64	0.0	0.0	0.0	0.0
Residual errors for finding all multiple roots							
$\text{SSM}^{\Gamma^{[*]}}$	0.0633	2.1*e*-35	6.8*e*-55	4.1*e*-33	0.0	0.0	0.0
$\text{SSM}^{\Gamma^{[1]}}$	0.0323	0.0	0.0	4.1*e*-33	0.0	0.0	0.0
$\text{SSM}^{\Gamma^{[2]}}$	0.0333	0.0	0.0	4.1*e*-33	0.0	0.0	0.0
$\text{SSM}^{\Gamma^{[3]}}$	0.0323	0.0	0.0	0.0	0.0	0.0	0.0

$\text{SSM}^{\partial^{[1]}}$–$\text{SSM}^{\partial^{[3]}}$, $\text{SSM}^{\partial^{[*]}}$ as

$$\begin{bmatrix} \overset{(0)}{\varsigma_1} = -1.1 + 0.2i, \\ \overset{(0)}{\varsigma_2} = -2.1 - 0.2i, \\ \overset{(0)}{\varsigma}_3 = 0.8 + 1.2i, \\ \overset{(0)}{\varsigma}_4 = 0.9 - 1.2i \end{bmatrix}.$$

For distinct root

$$\partial_*(\varsigma) = \begin{bmatrix} (\varsigma + 1)(\varsigma + 2) \\ (\varsigma - 1 - i) \\ (\varsigma - 1 + i) \end{bmatrix}.$$

The outcomes of the problems using numerical schemes are shown in Table 5. Table 5 clearly presents the dominance convergence rate of the scheme $\text{SSM}^{\partial^{[1]}}$–$\text{SSM}^{\partial^{[3]}}$ over existing method $\text{SSM}^{\partial^{[*]}}$.

Example 5. Consider a problem of beam positioning, resulting a nonlinear polynomial equation given as

$$\partial_3(\varsigma) = \frac{\varsigma^3 + 2.87s^2 - 10.28}{4.62} - \varsigma. \tag{57}$$

The exact roots of (57) are $\xi_1 = 2.0021$, $\xi_2 = -3.3304$, $\xi_3 = -1.5417$.

The initial estimates for $\partial_3(\varsigma)$ are

$$\begin{bmatrix} \overset{(0)}{\varsigma_{1,2}} = 2.5, \ \overset{(0)}{\varsigma_3} = -7.4641, \\ \overset{(0)}{\varsigma}_4 = -0.5359 \end{bmatrix}.$$

The outcomes of the problems using numerical schemes are shown in Table 6. Table 6 clearly presents the dominance convergence rate of the scheme $\text{SSM}^{\partial^{[1]}}$–$\text{SSM}^{\partial^{[3]}}$ over existing method $\text{SSM}^{\partial^{[*]}}$.

Example 6. Consider

$$\partial(\varsigma) = \begin{bmatrix} \sin^3\left(\frac{\varsigma - 1}{2}\right) \sin^3\left(\frac{\varsigma - 2}{2}\right) \\ \sin^3\left(\frac{\varsigma - 2.5}{2}\right) \end{bmatrix} \tag{58}$$

with exact roots $\xi_1 = 1$, $\xi_2 = 2$, $\xi_3 = 2.5$, and we choose the following initial guess values to approximate all roots of (56) using scheme $\text{SSM}^{\partial^{[1]}}$–$\text{SSM}^{\partial^{[3]}}$, $\text{SSM}^{\partial^{[*]}}$ as

$$\overset{(0)}{\varsigma_1} = -0.2, \overset{(0)}{\varsigma_2} = 1.7, \overset{(0)}{\varsigma}_3 = 3.$$

For distinct roots,

$$\partial(\varsigma) = \begin{bmatrix} \sin\left(\frac{\varsigma - 1}{2}\right) \sin\left(\frac{\varsigma - 2}{2}\right) \\ \sin\left(\frac{\varsigma - 2.5}{2}\right) \end{bmatrix}. \tag{59}$$

The outcomes of the problems using numerical schemes are shown in Table 7. Table 7 clearly presents the dominance convergence rate of the scheme $\text{SSM}^{\partial^{[1]}}$–$\text{SSM}^{\partial^{[3]}}$ over existing method $\text{SSM}^{\partial^{[*]}}$.

5. CONCLUSION

In this study, we developed a family of single-root finding methods and extended them into efficient two-step parallel simultaneous schemes for computing all roots of nonlinear equations. The percentage computational efficiency curves (Figs. 1a–1d) and the count of basic operations (Table 1) demonstrate that the proposed methods $\text{SSM}^{\partial^{[1]}}$–$\text{SSM}^{\partial^{[3]}}$

outperform the existing method $\mathrm{SSM}^{\partial[*]}$. Performance evaluation across various engineering problems shows that the new methods achieve lower residual errors, fewer iterations, and reduced CPU time compared to $\mathrm{SSM}^{\partial[*]}$ (Tables 2–7), confirming their effectiveness and consistency.

Limitations

- This study focuses primarily on nonlinear equations with well-separated roots; performance with highly clustered or nearly multiple roots requires further investigation.
- The methods have been tested mainly for low to moderate-dimensional problems; scalability for very large-scale systems is not yet assessed.
- Fractional-order and stochastic differential equations have not been considered in the current framework.

Future Work

- Higher-order parallel approaches will be developed to further increase convergence rates and computation efficiency.
- Extension of the proposed methods to handle clustered, nearly multiple, or complex-root systems.
- Application of the schemes to fractional, stochastic, and high-dimensional nonlinear problems in engineering and biomedical models.
- Integration with GPU-based or parallel computing architectures for large-scale simulations.

ACKNOWLEDGMENTS

This work was supported and funded by the deanship of scientific research at Imam Mohammad Ibn Saud Islamic University (IMSIU) (grant number IMSIU-DDRSP2603).

ORCID

Mudassir Shams
https://orcid.org/0000-0002-2980-5801
Nasreen Kausar
https://orcid.org/0000-0002-8659-0747
Praveen Agarwal
https://orcid.org/0000-0001-7556-8942
Nouf Abdulrahman Alqahtani
https://orcid.org/0009-0003-6771-0525
Mdi Begum Jeelani
https://orcid.org/0000-0002-8812-2859

REFERENCES

1. F. Wu, K. Zhang, J. Zhao, Y. Jin and D. Li, Linear and nonlinear GNSS PWV features for heavy rainfall forecasting, *Adv. Space. Res.* **72**(6) (2023) 2170–2184.
2. H. K. Chao, Three kinds of the Lotka–Volterra model transfer from biology to economics, *Synthese* **202**(4) (2023) 124.
3. D. Vignesh, S. He and S. Banerjee, A review on the complexities of brain activity: Insights from nonlinear dynamics in neuroscience, *Nonlinear Dyn.* **113**(5) (2025) 4531–4552.
4. D. G. Clark and L. F. Abbott, Theory of coupled neuronal-synaptic dynamics, *Phys. Rev. X* **14**(2) (2024) 021001.
5. F. I. Chicharro, A. Cordero and J. R. Torregrosa, Drawing dynamical and parameters planes of iterative families and methods, *Sci. World J.* (2013), https://doi.org/10.1155/2013/780153.
6. R. King, Family of fourth-order methods for nonlinear equations, *SIAM J. Numer. Anal.* **10**(5) (1973) 876–879.
7. A. Cordero, J. L. Hueso, E. Mart ínez and J. R. Torregrosa, New modifications of Potra–Pták's method with optimal fourth and eighth orders of convergence, *J. Comput. Appl. Math.* **234**(10) (2010) 2969–2976.
8. C. Chun, Some fourth-order iterative methods for solving nonlinear equations, *Appl. Math. Comput.* **195**(2) (2008) 454–459.
9. R. Behl, V. Kanwar and K. K. Sharma, Modified optimal families of fourth-order Jarratt's method, *Int. J. Pure Appl. Math.* **84**(4) (2013) 331–343.
10. J. Kou, Y. Li and X. Wang, A composite fourth-order iterative method for solving non-linear equations, *Appl. Math. Comput.* **184**(2) (2007) 471–475.
11. C. Chun and Y. M. Ham, Some fourth-order modifications of Newton's method, *Appl. Math. Comput.* **197** (2008) 654–658.
12. P. Jarratt, Some efficient fourth-order multipoint methods for solving equations, *BIT Numer. Math.* **9** (1969) 119–124.
13. F. I. Chicharro, A. Cordero, N. Garrido and J. R. Torregrosa, Generating root-finder iterative methods of second order: Convergence and stability, *Axioms* **8** (2019) 55.
14. A. M. Ostrowski, *Solution of Equations and Systems of Equations* (Prentice-Hall, Englewood Cliffs, NJ, USA, 1964).
15. A. Cordero, J. García-Maimó, J. R. Torregrosa, M. P. Vassileva and P. Vindel, Chaos in King's iterative family, *Appl. Math. Lett.* **26** (2013) 842–848.
16. O. Aberth, Iteration methods for finding all zeros of a polynomial simultaneously, *Math. Comput.* **27** (1973) 339–344.

17. M. Cosnard and P. Fraigniaud, Finding the roots of a polynomial on an MIMD multicomputer, *Parallel Comput.* **15**(1–3) (1990) 75–85.
18. P. D. Proinov, General convergence theorems for iterative processes and applications to the Weierstrass root-finding method, *J. Complexity* **33** (2016) 118–144
19. N. A. Mir, R. Muneer and I. Jabeen, Some families of two-step simultaneous methods for determining zeros of non-linear equations, *ISRN Appl. Math.* (2011), https://doi.org/10.5402/2011/817174.
20. S. Kanno, N. Kjurkchiev and T. Yamamoto, On some methods for the simultaneous determination of polynomial zeros,*Jpn. J. Appl. Math.* **13** (1995) 267–288.
21. A. W. Nourein, An improvement on two iteration methods for simultaneously determination of the zeros of a polynomial, *Int. J. Comput. Math.* **6** (1977) 241–252.
22. P. D. Proinov and M. T. Vasileva, On the convergance of higher-order Ehrlich-type iterative methods for approximating all zeros of polynomial simultaneously, preprint (2015), arXiv:1508.03359.
23. G. H. Nedzhibov, Iterative methods for simultaneous computing arbitrary number of multiple zeros of nonlinear equations,*Int. J. Comput. Math.* **90**(5) (2013), 994–1007.
24. S. I. Cholakov, Local and semilocal convergence of Wang–Zheng's method for simultaneous finding polynomial zeros, *Symmetry* **2019** (2019) 736.
25. M. S. Petković, L. D. Petkovi ć and J. Džunić, On an efficient method for the simultaneous approximation of polynomial multiple root, *Appl. Anal. Discrete Math.* **8** (2014) 73–94.
26. 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 (2014).
27. H. T. Kung and J. F. Traub, Optimal order of one-point and multipoint iteration, *J. Assoc. Comput. Mach.* **21** (1974) 643–651.
28. D. V. Griffithms and I. M. Smith, *Numerical Methods for Engineers*, 2nd edn. (Chapman and Hall/CRC (Taylor and Francis Group), 2011), Special Indian Edition.
29. B. Bradie, *A Friendly Introduction to Numerical Analysis* (Pearson Education, New Delhi, 2006).
30. P. D. Proinov, On the local convergence of Ehrlich method for numerical computation of polynomial zeros, *Calcolo* **53**(3) (2016) 413–426.