RESEARCH ARTICLE • OPEN ACCESS

# A Four-Step High-Order Iterative Method for Nonlinear Equations with Scientific Applications

Supriadi Putra et al.

JAMBURA
JOURNAL OF MATHEMATICS

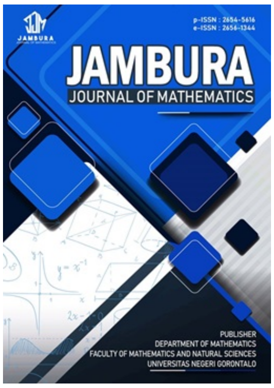
**Volume 8, Issue 1, Pages 52–59, February 2026**

*Received 5 November 2025, Revised 19 January 2026, Accepted 30 January 2026, Published 3 February 2026*

**To Cite this Article :** S. Putra et al.,“A Four-Step High-Order Iterative Method for Nonlinear Equations with Scientific Applications”, *Jambura J. Math*, vol. 8, no. 1, pp. 52–59, 2026, *https://doi.org/10.37905/jjom.v8i1.35161*



## JOURNAL INFO • JAMBURA JOURNAL OF MATHEMATICS




| | | |
|---|---|---|
| Homepage | : | http://ejurnal.ung.ac.id/index.php/jjom/index |
| Journal Abbreviation | : | Jambura J. Math. |
| Frequency | : | Biannual (February and August) |
| Publication Language | : | English (preferable), Indonesia |
| DOI | : | https://doi.org/10.37905/jjom |
| Online ISSN | : | 2656-1344 |
| Editor-in-Chief | : | Hasan S. Panigoro |
| Publisher | : | Department of Mathematics, Universitas Negeri Gorontalo |
| Country | : | Indonesia |
| OAI Address | : | http://ejurnal.ung.ac.id/index.php/jjom/oai |
| Google Scholar ID | : | iWLjgaUAAAAJ |
| Email | : | info.jjom@ung.ac.id |

**Jambura Journal of Mathematics**, Volume 8, Issue 1, Pages 52–59, February 2026 https://doi.org/10.37905/jjom.v8i1.35161

Research Article

Check for updates

# A Four-Step High-Order Iterative Method for Nonlinear Equations with Scientific Applications

**Supriadi Putra[1,*], Ayunda Putri[1], Zulkarnain Zulkarnain[1], Rike Marjulisa[1], Devi Novita[1]**

[1]*Department of Mathematics, Universitas Riau, Pekanbaru 28293, Indonesia*

**ARTICLE HISTORY**
Received 5 November 2025
Revised 19 January 2026
Accepted 30 January 2026
Published 3 February 2026

**KEYWORDS**
Four-step iterative method
Higher order of convergence
Predictor–corrector schemes
Polynomial interpolation
Derivative approximation

**ABSTRACT.** *In this paper, we propose a new four-step iterative method for solving nonlinear equations based on a predictor–corrector framework that combines Newton's, Ostrowski's, and Householder's methods. To avoid explicit evaluation of higher derivatives, particularly the second derivative, polynomial interpolation is employed to approximate derivative information in the higher-order step, while retaining first-derivative evaluations where required. The resulting scheme attains an optimal convergence order of fourteen using six function evaluations per iteration. Numerical experiments on several benchmark functions and two classical application problems, namely the computation of libration points and a Fibonacci-type root-finding problem, demonstrate improved accuracy and robust convergence behavior. In the reported tests, the method achieves the expected computational order of convergence and typically converges within a small number of iterations. The convergence properties are further examined through residual errors, step differences, and the observed computational order of convergence.*



## 1. Introduction

Solving nonlinear equations is a fundamental task in numerical analysis with wide-ranging applications in science, engineering, and applied mathematics. These equations often arise in models describing physical phenomena, such as heat transfer, chemical reactions, electrical circuits, orbital mechanics, and population dynamics. In recent years, the urgency of accurate and efficient root-finding methods has grown, particularly due to their central role in computational simulations of complex systems. Finding the roots of nonlinear equations, for instance, is crucial in some real-life applications of civil and chemical engineering. Some more examples of these applications are discussed in [1–3]. Problems investigated in these studies heavily rely on solving nonlinear algebraic systems, where in these contexts, the precision and reliability of root approximations can directly influence the success or failure of simulations, designs, or real-time decision-making processes.

Due to the lack of general closed-form solutions, iterative methods remain the primary computational tool for approximating the roots of nonlinear equations of the form

$$f(x) = 0 \tag{1}$$

where $f : \mathbb{R} \longrightarrow \mathbb{R}$ is a sufficiently smooth function. Among the most well-known iterative methods is Newton's method, which is defined as

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)},$$

for $i = 0, 1, 2, \ldots$, which is widely appreciated for its quadratic convergence near simple roots [4]. However, its practical implementation relies on the availability and accurate computation of the derivative $f'(x)$, which can be difficult, expensive, or even infeasible in certain applications. Furthermore, Newton's method may exhibit poor performance or divergence when applied with an initial guess far from the root or in the presence of singularities.

To overcome these limitations, several higher-order methods have been proposed in the literature. These include Ostrowski's method,

$$\begin{aligned} y_i &= x_i - \frac{f(x_i)}{f'(x_i)} \\ x_{i+1} &= y_i - \frac{f(y_i)(x_i - y_i)}{f(x_i) - 2f(y_i)}, \end{aligned}$$

which improves convergence order through additional correction steps [5]. This method is proven to be of order four. Householder's method, which achieves cubic or higher-order convergence by incorporating higher-order derivatives [6] is defined as:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} - \frac{f(x_i)^3 f''(x_i)}{2f'(x_i)^3}.$$

Despite their theoretical advantages, these methods suffer from a growing reliance on derivative evaluations, which can significantly increase computational cost and limit applicability.

An active area of research therefore focuses on the construction of high-order iterative methods that reduce the reliance on higher derivatives while maintaining computational efficiency and practical robustness. In particular, considerable

*Corresponding Author.

Email : *supriadiputra@lecturer.unri.ac.id* (S. Putra)
Homepage : http://ejurnal.ung.ac.id/index.php/jjom/index / E-ISSN : 2656-1344

attention has been given to schemes that approximate higher derivatives using interpolation or finite-difference techniques in order to simplify implementation. For instance, in [7], a cubic interpolation polynomial is employed to approximate the second derivative in the Householder method, leading to a higher-order scheme free from explicit second-derivative evaluation. Modified families of fourth- and eighth-order methods based on rational interpolation and Padé approximation are developed in [8], while [9] proposes polynomial interpolation and divided differences to construct fifth, ninth, and tenth-order methods. In addition, forward and finite-difference strategies are integrated into a modified Householder framework in [10, 11]. Another promising approach is to blend classical predictor-corrector structures with polynomial interpolation, forward difference, or finite difference techniques, enabling much high-order convergence without the explicit use of second or higher derivatives, such as in [12–17].

Despite these advances, many existing approaches still achieve relatively moderate orders of convergence or exhibit sensitivity to the choice of initial approximations, which can limit their robustness in practical applications. This indicates that there remains scope for the development of higher-order iterative schemes that both avoid explicit higher-derivative evaluations and demonstrate improved convergence behavior across a wider range of initial guesses.

In this work, we address these limitations by proposing a four-step iterative method that integrates Newton's, Ostrowski's, and Householder's methods within a predictor–corrector framework. The objective is to achieve a very high order of convergence while keeping the computational cost per iteration low and avoiding explicit evaluation of higher derivatives. In the higher-order step, polynomial interpolation is employed to approximate derivative information, thereby eliminating the need for direct second-derivative evaluations without introducing additional derivative computations at multiple internal points. The resulting scheme attains fourteenth-order convergence using a limited number of function evaluations per iteration. Numerical and theoretical analyses indicate that the method maintains robust convergence behavior, including reduced sensitivity to initial approximations, when compared with several existing high-order schemes.

The structure of this paper is as follows. In the next section, we present the construction of the proposed method, including the strategy used to approximate the first and second derivatives. We will also describe the theoretical order of convergence of the method. The numerical experiments are then provided in the subsequent section, which is divided into three parts: tests on benchmark nonlinear functions, an application to computing *libration* points in the Sun–Earth system, and an application involving a classical algebraic equation attributed to Fibonacci. Finally, the paper concludes with a summary of findings and suggestions for future research directions.

## 2. Methods

In this section, we develop the proposed method, discuss the approximation of the first and second derivatives, and provide a theoretical result establishing its order of convergence.

### 2.1. Construction of the Proposed Method

The proposed method is designed by combining Ostrowski's method as the predictor, and Householder method on the third step. Finally, we use Newton's method as the corrector on the final stage of the proposed multistep method. The method is given as follows:

$$y_i = x_i - \frac{f(x_i)}{f'(x_i)}, \tag{2}$$

$$z_i = y_i - \frac{f(y_i)(x_i - y_i)}{f(x_i) - 2f(y_i)}, \tag{3}$$

$$w_i = z_i - \frac{f(z_i)}{f'(z_i)} - \frac{f(z_i)^3 f''(z_i)}{2f'(z_i)^3}, \tag{4}$$

$$x_{i+1} = w_i - \frac{f(w_i)}{f'(w_i)}, \tag{5}$$

for $i = 0, 1, 2, \dots$.

To improve the efficiency, we introduce approximation to the derivatives, by considering $f(x) = \xi(x) = \beta_0 + \beta_1(x - y_i) + \beta_2(x - y_i)^2$ and imposing the following interpolating conditions:

$$f'(x_i) = \xi'(x_i), \quad , f(y_i) = \xi(y_i), \quad , f(z_i) = \xi(z_i).$$

By differentiating $\xi$, we obtain

$$\xi'(x) = \beta_1 + 2\beta_2(x - y_i), \tag{6}$$

$$\xi''(x) = 2\beta_2. \tag{7}$$

From the interpolating conditions, we obtain a system of nonlinear equations as follows:

$$\beta_1 + 2\beta_2(x_i - y_i) = f'(x_i), \tag{8}$$

$$\beta_0 + \beta_1(y_i - y_i) + \beta_2(y_i - y_i)^2 = f(y_i), \tag{9}$$

$$\beta_0 + \beta_1(z_i - y_i) + \beta_2(z_i - y_i)^2 = f(z_i). \tag{10}$$

By solving system (8)-(9) and utilizing eq. (6) and eq. (7), we have the approximations for $f'(z_i)$ and $f''(z_i)$ as follow:

$$f'(z_i) = \frac{2f[z_i, y_i](x_i - z_i) + (z_i - y_i)f'(x_i)}{2x_i - z_i - y_i}, \tag{11}$$

$$f''(z_i) = \frac{4(f'(x_i) - f[z_i, y_i])}{2x_i - y_i - z_i}, \tag{12}$$

with $f[z_i, y_i] = \dfrac{f(z_i) - f(y_i)}{z_i - y_i}$. This concludes the construction the proposed method, which is described by eq. (2)–(5), eq. (11), and eq. (12). In the following section, this method will be proven of order at least 14th.

### 2.2. Order of Convergence

**Theorem 1.** *Let $\alpha \in \mathcal{D}$ be a simple root of eq. (1) where function $f : \mathcal{D} \subset \mathbb{R} \longrightarrow \mathbb{R}$, where $f(x)$ is sufficiently in an open interval $\mathcal{D}$. If $x_0$ be an initial guess that properly close to $\alpha$, then the method described in eq. (2)–(5), eq. (11), eq. (12), which later will be referred to as M14, has at least of fourteenth order of convergence.*

*Proof.* Let $\alpha$ be a simple root of $f(x)$. Expanding $f(x)$ around $x=\alpha$ using Taylor expansion gives

$$f(x) = f(\alpha) + (x-\alpha)f'(\alpha) + \frac{1}{2!}(x-\alpha)^2 f''(\alpha) + \frac{1}{3!}(x-\alpha)^3 f^{(3)}(\alpha) + \frac{1}{4!}(x-\alpha)^4 f^{(4)}(\alpha) + \dots \tag{13}$$

Followed by evaluating eq. (13) at $x_i$, we obtain

$$f(x_i) = C_1\big(e_i + C_2e_i^2 + C_3e_i^3 + C_4e_i^4 + C_5e_i^5 + C_6e_i^6 + \dots\big), \tag{14}$$

where $C_j = \frac{1}{j!}\frac{1}{f'(\alpha)}D^{(j)}(f)(\alpha)$, and $e_i = x_i - \alpha$ for $i = 0,1,2,\dots$. Differentiating eq. (13) and evaluating it at $x_i$ yields

$$f'(x_i) = C_1\big(1 + 2C_2e_i + 3C_3e_i^2 + 4C_4e_i^3 + 5C_5e_i^4 + 6C_6e_i^5 + \dots\big). \tag{15}$$

Substituting eq. (14) and eq. (15) into eq. (2) and simplifying gives

$$y_i = \alpha + C_2e_i^2 + (-2C_2^2 + 2C_3)e_i^3 + \big(-4C_2^3 - 7C_2C_3 + 3C_4\big)e_i^4 + \dots. \tag{16}$$

Evaluating $f(x)$ at $y_i$, we have

$$f(y_i) = C_1\big(C_2e_i^2 + (-2C_2^2 + 2C_3)e_i^3 + (-3C_2^3 - 7C_2C_3 + 3C_4)e_i^4 + \dots\big). \tag{17}$$

Consequently,

$$z_i = \alpha + (2C_2^3 - C_2C_3)e_i^4 + \big(-8C_2^4 + 17C_2^2C_3 - 2C_2C_4 - 2C_3^2\big)e_i^5 + \dots. \tag{18}$$

Next, by calculating $f(z_i)$, with the same manner as in eq. (14), we have

$$f(z_i) = C_1\big((2C_2^3 - C_2C_3)e_i^4 + (-8C_2^4 + 17C_2^2C_3 - 2C_2C_4 - 2C_3^2)e_i^5 + \dots\big) \tag{19}$$

Using eq. (17) and eq. (19), we have $f[z_i, y_i]$ as follows:

$$f[z_i,y_i] = C_1 + C_1C_2^2e_i^2 + \left(\frac{8C_1C_3^3}{C_2^3} - 10C_1C_2^3 + 26C_1C_2C_3 - \frac{24C_1C_3^2}{C_2}\right)e_i^3 + \dots. \tag{20}$$

Eq. (20) with eq. (15) are used to approximate $f'(z_i)$ in eq. (11) as follows:

$$f'(z_i) = C_1 + \left(\frac{8C_1C_3^2}{C_2^3} - \frac{65C_1C_2^3}{8} + \frac{45C_1C_2C_3}{2} - \frac{24C_1C_3^2}{C_2}\right)e_i^3 + \left(\frac{36C_1C_3^2C_4}{C_2^3} - \frac{72C_1C_2C_4}{C_2} - \frac{68C_1C_3^2}{C_2^2}\right)e_i^4 + \dots. \tag{21}$$

The approximation to $f''(z_i)$ is done by substituting $x_i$, eq. (15), eq. (16), eq. (18), and eq. (20) into eq. (12) which resulting in

$$f''(z_i) = 4C_1C_2 + 6C_1C_3e_i + \Big(16C_2C_2^3 + 8C_4C_1 + \frac{48C_1C_3^2}{C_2} - \frac{16C_1C_3^3}{C_2^3} - 45C_1C_2C_3\Big)e_i^2 + \Big(10C_1C_5 - 114C_1C_3^2 - 68C_1C_2C_4 - \frac{349C_1C_2^2C_3}{2} + \dots\Big)e_i^3 + \dots. \tag{22}$$

Substituting eq. (18), eq. (19), eq. (21), and eq. (22), into eq. (4) and simplifying, we obtain

$$w_i = \alpha + \left(-\frac{65C_2^6}{4} + \frac{425C_2^4C_3}{8} - \frac{141C_2^2C_3^2}{2} + 40C_3^3 - \frac{8C_3^4}{C_2^2}\right)e_i^7 + \dots. \tag{23}$$

Next, expansion of $f(x)$ around $w_i$ yields

$$f(w_i) = \left(-\frac{65C_1C_2^6}{4} + \frac{425C_1C_2^4C_3}{8} - \frac{141C_1C_2^2C_3^2}{2} + 40C_1^3 - \frac{8C_1C_3^4}{C_2^2}\right)e_i^7 + \dots. \tag{24}$$

By evaluating eq. (23) at $x = w_i$, we obtain

$$f'(w_i) = C_1\Bigg(1 + 2C_2\left(-\frac{65C_2^6}{4} + \frac{452C_2^4C_3}{8} - \frac{141C_2^2C_3}{2} + 40C_3^3 - \frac{8C_3^4}{C_2^2}\right)e_i^7 + 3C_3\left(-\frac{65C_2^6}{4} + \frac{425C_2^4C_3}{8} - \frac{141C_2^2C_3^2}{2} + 40C_3^3 - \frac{8C_3^4}{C_2^2}\right)e_i^{14} + \dots\Bigg). \tag{25}$$

Finally, by inserting eq. (24) and eq. (25) into eq. (5), we show that

$$x_{i+1} = \alpha + \left(-\frac{2762C_2^{11}C_3}{8} + \frac{327265C_2^9C_3^2}{32} - \frac{70325C_2^7C_3^3}{4} + \frac{128C_3^8}{C_2^3} + \frac{37921C_2^5C_3^4}{2} - 12980C_2^3C_3^5 + 5456C_2C_3^6 - \frac{1280C_3^7}{C_2} + \frac{4225C_2^{13}}{8}\right)e_i^{14} + \dots,$$

which concludes the proof. □

## 3. Results and Discussion

In this section, we test the proposed methods on several functions and compare the results with several methods of higher order where we impose several metrics to asses the observation. In addition, to further evaluate the performance and robustness of the proposed high-order iterative method, we apply it to two nonlinear problems with historical and scientific significance. These real-world and classical cases offer a valuable benchmark for assessing not only convergence speed and accuracy but also stability in solving ill-conditioned or sensitive equations.

### 3.1. Numerical Results on Several Test Functions

In this part, we apply our proposed method to several test functions to evaluate its performance. We compare the results

**Table 1.** Numerical results for the test functions, $f_1$ and $f_2$

| | | | Iter | ACOC | $\lvert f(x_{i+1})\rvert$ | $\lvert x_{i+1}-x_i\rvert$ |
|---|---|---|---|---|---|---|
| $f_1(x)$ | $x_0=0.1$ | NM | 16 | 2.0000 | 1.774e-281 | 2.432e-141 |
| | | OM | 11 | 4.0000 | 4.864e-839 | 2.221e-210 |
| | | HM | 30 | 3.0000 | 7.218e-582 | 1.130e-194 |
| | | MKong12 | 8 | 11.9733 | 0.000e+00 | 5.104e-143 |
| | | MPairat12 | - | - | - | - |
| | | MUllah16 | 8 | 16.3896 | 0.000e+00 | 2.982e-165 |
| | | M14 | 5 | 13.6473 | 4.034e-881 | 1.409e-63 |
| | $x_0=1.4$ | NM | 12 | 2.0000 | 4.471e-377 | 3.860e-189 |
| | | OM | 6 | 4.0000 | 3.375e-556 | 1.140e-139 |
| | | HM | 33 | 3.0000 | 2.071e-464 | 1.606e-155 |
| | | MKong12 | 4 | 11.8509 | 4.540e-857 | 4.206e-72 |
| | | MPairat12 | 5 | 4.0000 | 2.827e-352 | 1.435e-88 |
| | | MUllah16 | 19 | 13.2851 | 1.160e-607 | 1.231e-36 |
| | | M14 | 3 | 15.2002 | 5.806e-391 | 1.446e-28 |
| $f_2(x)$ | $x_0=1.5$ | NM | 8 | 2.0000 | 1.325e-295 | 2.609e-148 |
| | | OM | 4 | 4.0000 | 6.992e-300 | 1.617e-75 |
| | | HM | 6 | 3.0000 | 1.193e-749 | 1.615e-250 |
| | | MKong12 | 3 | 11.9770 | 0.000e+00 | 2.461e-165 |
| | | MPairat12 | 4 | 4.0000 | 1.887e-342 | 4.916e-86 |
| | | MUllah16 | 2 | - | 1.024e-320 | 1.019e-20 |
| | | M14 | 3 | 13.9863 | 0.000e+00 | 1.612e-246 |
| | $x_0=-0.5$ | NM | 12 | 2.0000 | 1.941e-352 | 9.988e-177 |
| | | OM | 6 | 4.0000 | 3.809e-261 | 7.810e-66 |
| | | HM | 12 | 3.0000 | 2.484e-618 | 9.574e-207 |
| | | MKong12 | 4 | 10.6912 | 1.031e-454 | 1.684e-38 |
| | | MPairat12 | 6 | 4.0000 | 5.986e-768 | 2.075e-192 |
| | | MUllah16 | 22 | 12.8641 | 8.070e-736 | 1.159e-46 |
| | | M14 | 4 | 13.9477 | 0.000e+00 | 1.554e-176 |
| $f_3(x)$ | $x_0=-2.0$ | NM | 12 | 2.0000 | 8.524e-326 | 5.286e-164 |
| | | OM | 16 | 4.0000 | 2.165e-680 | 7.270e-171 |
| | | HM | 8 | 3.0000 | 4.157e-458 | 9.342e-154 |
| | | MKong12 | 4 | 13.0603 | 2.348e-599 | 1.193e-50 |
| | | MPairat12 | 5 | 4.0000 | 8.353e-389 | 1.099e-97 |
| | | MUllah16 | - | - | - | - |
| | | M14 | 4 | 14.5862 | 1.071e-675 | 4.447e-49 |
| | $x_0=-10.0$ | NM | - | - | - | - |
| | | OM | 47 | 4.0000 | 5.530e-806 | 2.906e-202 |
| | | HM | 73 | 3.0000 | 1.226e-264 | 2.886e-89 |
| | | MKong12 | 44 | 12.1460 | 1.200e-998 | 7.528e-122 |
| | | MPairat12 | 34 | 4.0000 | 1.134e-462 | 3.750e-116 |
| | | MUllah16 | - | - | - | - |
| | | M14 | 80 | 14.6431 | 8.954e-643 | 9.993e-47 |

with several other methods such as Newton's method (NM), Ostrowski method (MO), Householder's method (MH), a 12th order method by Kong-ied [14] (MKong12), another 12th order method by Janngam and Comemuangb [13] (MPairat12), and a 16th order method by Ullah et al. [15] (MUllah16). The maximum number of iterations is set to $100$, with computations carried out using $1000$ significant digits and an error tolerance of $10^{-250}$. We consider the following three test functions.

- $f_1(x) = (x-1)^3 - 1$,
- $f_2(x) = \sin(x)^2 - x^2 + 1$,
- $f_3(x) = x\exp(x^2) - \sin(x)^2 + 3\cos(x) + 5$.

Based on the numerical results presented in the following tables, we can observe and analyze the performance of the proposed method when applied to the test functions. The metrics used to evaluate performance include the number of iterations (Iter), the approximated computational order of convergence (ACOC), the absolute function value at the final iterate $|f(x_{i+1})|$, and the difference between successive iterates $|x_{i+1}-x_i|$. Methods that fail to converge within the maximum number of iterations or yield unstable results are denoted by –.

Table 1 reveals that all converging methods achieve very high residual precision, frequently below $10^{-800}$ and extremely small update steps $|x_{i+1}-x_i|$, some as low as $10^{-210}$. In the case of $f_1$, NM consistently converged in 12-16 iterations with second-order behavior (ACOC $= 2.0000$). In addition, Householder's method (HM), though of higher order (third-order), required $30$–$33$ iterations, which shows either poor damping or slower local convergence. Among higher-order methods, M14 consistently showed excellent performance, converging in just 3–5 iterations across both initial guesses. It achieved residuals

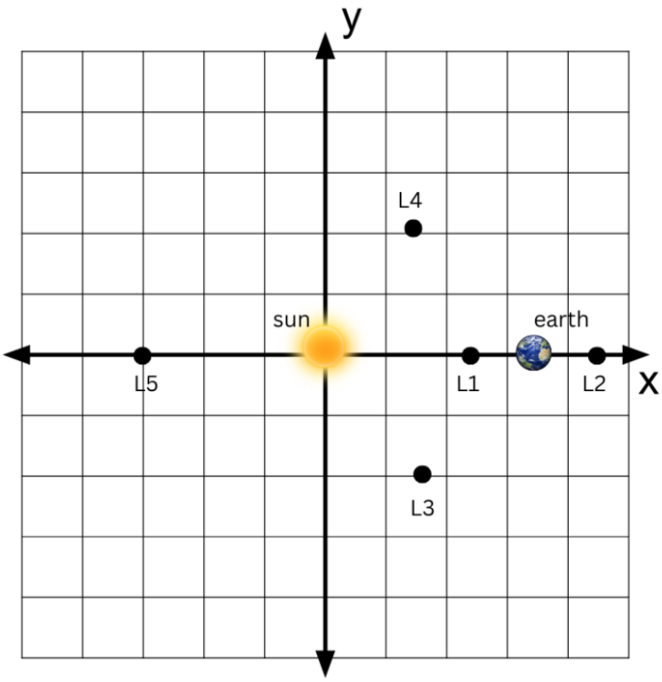



**Figure 1.** Libration points of a satellite relative to the rotating Sun-Earth System

on the order of $10^{-800}$ or smaller and inter-iterate differences well below $10^{-50}$, confirming very fine precision. Moreover, its ACOC values ranged from $13.6$ to $15.2$, showing consistent high-order convergence. Furthermore, MKong12 and MUllah16 also performed well on this problem, with convergence in $4$–$8$ iterations and high ACOC ($\geq 11.8$). In contrast, MPairat12 failed to converge for the initial guess $x_0 = 0.1$ likely due to sensitivity to the local behavior of $f_1$. This illustrates a fundamental limitation of certain high-order methods: although they are efficient near the root, they may be fragile to distant or poorly chosen starting points.

In the case of $f_2$, all methods successfully converged for both initial values $x_0 = 1.5$ and $x_0 = -0.5$, demonstrating that this function is more numerically well-conditioned for root-finding algorithms. NM maintained its expected behavior, converging in 12 iterations with ACOC $= 2$. OM and HM needed only $4$–$12$ iterations depending on the initial guess. Meanwhile, high-order methods such as M14, MKong12, MPairat12, and MUllah16 exhibited significantly improved performance. M14 converged in just $3$–$4$ iterations with ACOC $\approx 14.0$, consistently producing residuals under $10^{-800}$ and very small step sizes. The ACOC values remained stable across both initial values, suggesting robustness of the convergence rate for smooth functions. While no method failed to converge on $f_2$, minor variations in ACOC among higher-order methods (ranging from $10.7$ to $14.0$) reflect slight differences in step construction and numerical stability. Nonetheless, the residuals and update magnitudes confirm all methods attained high accuracy.

The function $f_3$ poses a much more challenging problem due to its combination of exponential and trigonometric terms. As shown in Table 1, convergence behavior depends strongly on the initial guess. For $x_0 = -2.0$, NM converged in 12 iterations, but higher-order methods clearly outperformed it. M14 again achieved convergence in just $4$ iterations with ACOC $\approx 14.5862$, final residual $\approx 10^{-675}$ and step size $\approx 10^{-49}$ . MKong12 and MPairat12 also performed very well. However, MUllah16 failed to converge for this initial value, possibly due to poor stability or oscillatory behavior in its iteration formula. Furthermore, with the more distant initial guess $x_0 = -10.0$ the convergence difficulty increased significantly. Both NM and MUllah16 failed to converge, reflecting their sensitivity to initial conditions and the non-polynomial structure of $f_3$. HM and OM required $47$ and $73$ iterations respectively, with moderate precision. Despite the increased complexity, M14 remained stable and converged in $80$ iterations, producing an excellent residual $\approx 10^{-643}$, a very small update size, and a stable ACOC $\approx 14.6431$. MKong12 and MPairat12 also converged but with slightly lower efficiency, needing $34$–$44$ iterations. These results highlight M14's advantage in maintaining both convergence and precision under extreme initial conditions where other methods struggle.

### *3.2. Applications to Libration Points in the Sun-Earth System and Fibonacci's Algebraic Challenge*

#### 3.2.1. Case 1: Libration Points in the Sun-Earth System

In this case, we consider a physical scenario involving the sun at the origin and the earth located at the point $(1, 0)$ with the distance measured in astronomical units (AU), where $1\text{AU} < 1.496 \times 10^8$ km. There exist five points, known as *libration* points ($L_1, L_2, L_3, L_4, L_5$), where a satellite can remain in equilibrium relative to the rotating Sun-Earth system. These points are of interest because the gravitational attractions from both celestial bodies balance the centrifugal force acting on the satellite [18].

Assuming $m_1$ and $m_2$ are the masses of the sun and the earth respectively, and defining the mass ratio as $r = \frac{m_1}{m_1+m_2}$, the $x-$coordinate of $L_1$ is the unique real root of the equation:

$$\theta_1 = x^5 - (2+r)x^4 + (1+2r)x^3 - (1-r)x^2 + 2(1-r)x + r - 1 = 0.$$

**Table 2.** Numerical results for the test functions, $\theta_1$ and $\theta_2$

| | | | Iter | ACOC | $\lvert f(x_{i+1})\rvert$ | $\lvert x_{i+1}-x_i\rvert$ |
|---|---|---|---|---|---|---|
| $\theta_1(x)$ | $x_0 = 0.0$ | NM | 19 | 2.0000 | 3.468e-415 | 1.982e-207 |
| | | OM | 9 | 4.0000 | 9.703e-266 | 1.140e-67 |
| | | HM | 13 | 3.0000 | 1.936e-589 | 2.371e-197 |
| | | MKong12 | 6 | 11.9800 | 0.000e+00 | 1.820e-165 |
| | | MPairat12 | - | - | - | - |
| | | MUllah16 | 5 | 15.0127 | 1.325e-833 | 1.879e-54 |
| | | M14 | 8 | 13.9746 | 0.000e+00 | 1.883e-187 |
| | $x_0 = -1.0$ | NM | 21 | 2.0000 | 5.516e-351 | 2.499e-175 |
| | | OM | 10 | 4.0000 | 2.045e-289 | 1.374e-73 |
| | | HM | 14 | 3.0000 | 5.473e-375 | 7.223e-126 |
| | | MKong12 | - | - | - | - |
| | | MPairat12 | - | - | - | - |
| | | MUllah16 | - | - | - | - |
| | | M14 | 7 | 12.5841 | 8.896e-420 | 3.181e-32 |
| $\theta_2(x)$ | $x_0 = 0.0$ | NM | 30 | 2.0000 | 3.716e-387 | 2.004e-193 |
| | | OM | 14 | 4.0000 | 8.051e-400 | 3.373e-101 |
| | | HM | 15 | 3.0000 | 1.055e-427 | 1.896e-143 |
| | | MKong12 | 12 | 12.0383 | 1.597e-999 | 5.700e-133 |
| | | MPairat12 | - | - | - | - |
| | | MUllah16 | 6 | 10.4458 | 2.094e-487 | 8.001e-33 |
| | | M14 | 5 | 13.5992 | 9.640e-821 | 7.127e-61 |
| | $x_0 = -1.0$ | NM | 30 | 2.0000 | 8.179e-450 | 9.401e-225 |
| | | OM | 12 | 4.0000 | 2.777e-497 | 1.454e-125 |
| | | HM | 52 | 3.0000 | 2.224e-392 | 1.128e-131 |
| | | MKong12 | - | - | - | - |
| | | MPairat12 | - | - | - | - |
| | | MUllah16 | - | - | - | - |
| | | M14 | 6 | 13.6226 | 1.560e-848 | 7.376e-63 |

Meanwhile, the $x$-coordinate of $L_2$ is the root of the quadratic equation:

$$\theta_2 - 2rx^2 = 0.$$

Using the value $r \approx 3.04042 \times 10^{-6}$, we compute numerical approximations of the coordinates of $L_1$ and $L_2$, and the results as shown in Table 2.

Table 2 shows the numerical results for the *libration* point equations $\theta_1$ and $\theta_2$ are reported for two different initial guesses. For $\theta_1$, classical methods such as Newton's (NM), Ostrowski's (OM), and Householder's (HM) require more iterations (ranging from 13 to 21) and demonstrate lower convergence orders (ACOC 2 to 4) compared to the higher-order methods. In contrast, M14 (the proposed 14th-order method) achieves convergence in only 7–8 iterations with an ACOC close to 14, and it successfully drives $|f(x_{i+1})|$ to nearly zero while maintaining an extremely small $|x_{i+1} - x_i|$. The methods MUllah16 and MKong12 methods also exhibit high convergence rates, confirming the superior accuracy of these advanced methods.

For the $\theta_2$ function, which is quadratic and simpler in structure, the iteration counts are lower across all methods. Still, NM remains relatively inefficient, requiring 30 iterations with a second-order convergence. In comparison, the proposed M14 method achieves full convergence in only 5–6 iterations for both starting points, with ACOC values around $13.6$. MKong12 and MUllah16 also perform well when data is available, but the consistent success of M14 with minimal iterations highlights its robustness and efficiency even on relatively simple equations.

#### 3.2.2. Case 2: Fibonacci's Algebraic Challenge

This case revisits a historical mathematical problem from 1224, in which Leonardo of Pisa, widely known as Fibonacci, tackled a challenge posed by John of Palermo during a royal court event before Emperor Frederick II [4]. The task was to find the root of the equation

$$\eta(x) = x^3 + 2x^2 + 10x = 20.$$

Fibonacci first demonstrated that this equation has no rational roots and lacks any classical irrational root of the Euclidean type—that is, it cannot be expressed in the form $a \pm \sqrt{b}$, $\sqrt{a} \pm \sqrt{b}$, $\sqrt{a \pm \sqrt{b}}$, $\sqrt{\sqrt{a} \pm \sqrt{b}}$, where $a$ and $b$ are rational numbers. Fibonacci proceeded by approximating the only real root using a technique believed to be similar to one developed by Omar Khayyam, which involves geometric constructions such as the intersection of a circle and a parabola. He represented the approximation using the base-60 (sexagesimal) number system as:

$$\phi = 1 + 22\left(\frac{1}{60}\right) + 7\left(\frac{1}{60}\right)^2 + 33\left(\frac{1}{60}\right)^4 + 4\left(\frac{1}{60}\right)^5 + 40\left(\frac{1}{60}\right)^6.$$

We examine the accuracy of Fibonacci's approximation by solving the equation numerically using various iterative methods. The corresponding results are presented in Table 3.

Table 3 presents the numerical results for the equation $\eta(x)$, with two initial guesses. All methods converge to a highly

**Table 3.** Numerical results for the test function $\eta(x)$

| | | | Iter | ACOC | $\|f(x_{i+1})\|$ | $\|x_{i+1} - x_i\|$ | $\|\phi - x_{i+1}\|$ |
|---|---|---|---|---|---|---|---|
| $\eta(x)$ | $x_0 = 0.0$ | NM | 10 | 2.0000 | 1.665e-396 | 5.222e-199 | 3.185e-11 |
| | | OM | 5 | 4.0000 | 1.282e-628 | 1.550e-157 | 3.185e-11 |
| | | HM | 6 | 3.0000 | 5.203e-296 | 2.738e-99 | 3.185e-11 |
| | | MKong12 | 3 | 13.5829 | 2.758e-364 | 1.459e-30 | 3.185e-11 |
| | | MPairat12 | 4 | 4.0000 | 2.231e-326 | 1.645e-81 | 3.185e-11 |
| | | MUllah16 | 3 | 19.2143 | 3.304e-347 | 6.934e-21 | 3.185e-11 |
| | | M14 | 3 | 13.9039 | 1.000e-998 | 4.523e-101 | 3.185e-11 |
| | $x_0 = -1.0$ | NM | 10 | 2.0000 | 6.444e-337 | 3.248e-169 | 3.185e-11 |
| | | OM | 5 | 4.0000 | 3.402e-389 | 1.112e-97 | 3.185e-11 |
| | | HM | 8 | 3.0000 | 9.758e-684 | 1.567e-228 | 3.185e-11 |
| | | MKong12 | 3 | 12.3926 | 7.112e-309 | 6.049e-26 | 3.185e-11 |
| | | MPairat12 | 5 | 4.0000 | 1.470e-867 | 8.333e-217 | 3.185e-11 |
| | | MUllah16 | 17 | 12.6044 | 3.654e-470 | 4.057e-28 | 3.185e-11 |
| | | M14 | 3 | 13.6956 | 1.518e-888 | 1.126e-63 | 3.185e-11 |

accurate approximation of the root $\beta$, matching Fibonacci's historical value up to at least 11 decimal places (as shown in the consistent value $|\beta - x_{i+1}| \approx 3.185 \times 10^{-11}$ across all methods). Among the higher-order methods, M14 and MUllah16 consistently exhibit the best performance. For $x_0 = 0.0$, M14 reaches machine-precision accuracy in just 3 iterations with an ACOC of 13.9, while MUllah16 achieves an even higher ACOC (19.2) in 3 iterations. When starting from $x_0 = -1.0$, the trends remain similar: M14 and MUllah16 still outperform others in terms of speed and precision. MKong12 and MPairat12 also show strong performance, though with slightly more iterations. These results further affirm the reliability and accuracy of the proposed method and its potential for solving complex nonlinear equations with minimal computational effort.

## 4. Conclusion

In this work, we have developed a four-step iterative method that combines classical predictor–corrector schemes with a polynomial interpolation strategy to approximate derivative information. The proposed method achieves fourteenth-order convergence, as established theoretically, while avoiding explicit evaluation of second and higher derivatives by interpolation and requiring only first-derivative evaluations at selected stages. For the tested benchmark functions and application problems, including the computation of libration points in the Sun–Earth system and Fibonacci's algebraic root-finding challenge, the numerical results demonstrate high accuracy with a small number of iterations, very low residual errors, and consistent values of the ACOC. In particular, the proposed method successfully converged for several distant initial guesses in these applications, whereas existing high-order methods such as Mullah16, MKong12, and MPairat12 failed to converge or exhibited strong sensitivity to the choice of initial approximations. These observations indicate improved robustness of the proposed scheme with respect to initial guesses in the considered problems. From a practical standpoint, the method requires six function evaluations per iteration, which makes the high convergence order achievable without excessive computational cost. Future research may focus on extending the proposed framework to systems of nonlinear equations and stiff problems, as well as adapting it to derivative-free or memory-efficient formulations. Additional directions include the investigation of adaptive interpolation strategies, dynamic step-size control, and parallel implementations.

**Author Contributions.** **Supriadi Putra and Ayunda Putri**: Conceptualization, methodology, formal analysis, supervision. **Zulkarnain**: Methodology, algorithm, original draft preparation. **Rike Marjulisa**: Writing–review and editing. **Devi Novita**: Visualization and project administration. All authors have reviewed and approved the final published version of this paper.

**Acknowledgement.** The authors would like to express their gratitude to Lembaga Penelitian dan Pengabdian kepada Masyarakat (LPPM) Universitas Riau for supporting this research, and to the editors and reviewers who have supported us in improving this manuscript.

**Funding.** This research is funded by DIPA UNRI 2025 scheme with contract number 38511/UN19.5.1.3/AL.04/2025.

**Conflict of interest.** The authors declare no conflict of interest.

**Data availability.** Not applicable.

## References

[1] A. Naseem, M. A. Rehman, and T. Abdeljawad, "Real-world applications of a newly designed root-finding algorithm and its polynomiography," *IEEE Access*, vol. 9, pp. 160 868–160 877, 2021, doi: 10.1109/ACCESS.2021.3131498.

[2] A. Naseem, M. Rehman, T. Abdeljawad, and Y.-M. Chu, "Novel iteration schemes for computing zeros of non-linear equations with engineering applications and their dynamics," *IEEE Access*, vol. 9, pp. 92 246–92 262, 2021, doi: 10.1109/ACCESS.2021.3091473.

[3] A. Putri and M. Imran, "A new second derivative free iterative method of fifth order of convergence and its applications," *Jurnal Matematika UNAND*, vol. 12, no. 4, pp. 283–292, 2023, doi: 10.25077/jmua.12.4.283-292.2023.

[4] J. D. Faires and R. L. Burden, *Numerical methods, 4th*. Cengage Learning, 2012.

[5] A. Ostrowski, "Solution of equations and system of equations," 1960.

[6] A. S. Householder, "The numerical treatment of a single nonlinear equation," *(No Title)*, 1970.

[7] A. Putri, M. Imran, and R. Marjulisa, "Modified householder method of fifth order of convergence and its dynamics on complex plane," *Journal of Fundamental Mathematics and Applications (JFMA)*, vol. 6, no. 2, pp. 188–198, 2023, doi: 10.14710/jfma.v6i2.20554.

[8] O. S. Solaiman, S. A. A. Karim, and I. Hashim, "Optimal fourth-and eighth-order of convergence derivative-free modifications of king's method," *Jour-*

*nal of King Saud University-Science*, vol. 31, no. 4, pp. 1499–1504, 2019, doi: 10.1016/j.jksus.2018.12.001.

[9] O. Ogbereyivwe, O. Izevbizua, and S. S. Umar, "Some high-order convergence modifications of the householder method for nonlinear equations," *Communications in Nonlinear Analysis*, vol. 11, no. 2, pp. 1–11, 2023, doi: 20.1001.1.23717920.2023.11.1.5.5.

[10] A. Naseem, M. Rehman, and T. Abdeljawad, "A novel root-finding algorithm with engineering applications and its dynamics via computer technology," *IEEE Access*, vol. 10, pp. 19 677–19 684, 2022, doi: 10.1109/ACCESS.2022.3150775.

[11] S. Syamsudhuha, M. Imran, A. Putri, L. Deswita, and R. Amelia, "A new three-step derivative free iterative method and its dynamics," *Journal of the Indonesian Mathematical Society*, vol. 30, no. 3, pp. 361–373, 2024.

[12] S. Thota, T. Gemechu, and A. A. Ayoade, "On new hybrid root-finding algorithms for solving transcendental equations using exponential and halley's methods," *Ural Mathematical Journal*, vol. 9, no. 1 (16), pp. 176–186, 2023, doi: 10.15826/UMJ.2023.1.016.

[13] P. Janngam and C. Comemuangb, "New twelfth order iterative method for solving nonlinear equations and their dynamical aspects," *Journal of Mathematics and Computer Science*, vol. 28, no. 1, pp. 52–59, 2023.

[14] B. Kong-ied, "Two new eighth and twelfth order iterative methods for solving nonlinear equations," *International Journal of Mathematics and Computer Science*, vol. 16, no. 1, pp. 333–344, 2021.

[15] M. Z. Ullah, A. Al-Fhaid, and F. Ahmad, "Four-point optimal sixteenth-order iterative method for solving nonlinear equations," *Journal of applied mathematics*, vol. 2013, no. 1, p. 850365, 2013, doi: 10.1155/2013/850365.

[16] S. Thota, M. M. Awad, P. Shanmugasundaram, and L. Rathour, "A derivative-free root-finding algorithm using exponential method and its implementation," *BMC Research Notes*, vol. 16, no. 1, p. 276, 2023, doi: 10.1186/s13104-023-06554-1.

[17] M. S. Bahgat and M. Hafiz, "Three-step iterative method with eighteenth order convergence for solving nonlinear equations," *International Journal of pure and applied Mathematics*, vol. 93, no. 1, pp. 85–94, 2014, doi: 10.12732/ijpam.v93i1.7.

[18] J. Stewart, D. Clegg, and S. Watson, *Calculus: early transcendentals*. Cengage, 2021.