mathematics

MDPI

Article

Enhanced Ninth-Order Memory-Based Iterative Technique for Efficiently Solving Nonlinear Equations

Shubham Kumar Mittal [1], Sunil Panday [1,*] and Lorentz Jäntschi [2]

[1] Department of Mathematics, National Institute of Technology Manipur, Langol, Imphal 795004, Manipur, India; shubhammehara2@gmail.com
[2] Department of Physics and Chemistry, Technical University of Cluj-Napoca, 103-105 Muncii Blvd., 400641 Cluj-Napoca, Romania; lorentz.jantschi@chem.utcluj.ro
* Correspondence: sunilpanday@hotmail.co.in

Abstract: In this article, we present a novel three-step with-memory iterative method for solving nonlinear equations. We have improved the convergence order of a well-known optimal eighth-order iterative method by converting it into a with-memory version. The Hermite interpolating polynomial is utilized to compute a self-accelerating parameter that improves the convergence order. The proposed uni-parametric with-memory iterative method improves its R-order of convergence from 8 to 8.8989. Additionally, no more function evaluations are required to achieve this improvement in convergence order. Furthermore, the efficiency index has increased from 1.6818 to 1.7272. The proposed method is shown to be more effective than some well-known existing methods, as shown by extensive numerical testing on a variety of problems.

Keywords: nonlinear equation; roots; efficiency index; iterative method; with-memory methods

MSC: 41A25; 65H05; 65D99

check for updates

Citation: Mittal, S.K.; Panday, S.; Jäntschi, L. Enhanced Ninth-Order Memory-Based Iterative Technique for Efficiently Solving Nonlinear Equations. *Mathematics* **2024**, *12*, 3490. https://doi.org/10.3390/math12223490

Academic Editors: Juan Ramón Torregrosa Sánchez, Alicia Cordero and Francisco I. Chicharro

Received: 4 October 2024
Revised: 3 November 2024
Accepted: 6 November 2024
Published: 8 November 2024

1. Introduction

Many complex problems in science and engineering involve nonlinear equations of the form $\zeta(x) = 0$, where $\zeta : D \subset \mathbb{R} \to \mathbb{R}$ is a scalar function defined over an open interval D. Solutions to these equations typically cannot be expressed in closed form. As traditional analytical methods are often insufficient for solving such equations, iterative numerical techniques have become indispensable, especially with advancements in computational technology that enable more efficient and accurate solutions. Newton's method [1] is a widely used iterative technique for approximating the simple root ξ of $\zeta(x) = 0$. It follows the iterative formula:

$$x_{n+1} = x_n - \frac{\zeta(x_n)}{\zeta'(x_n)}, \quad n = 0, 1, 2, \ldots \tag{1}$$

where ζ is the function and ζ' is its derivative. Newton's method is known for its quadratic convergence near the root and it requires the evaluation of both the function and its derivative in each iteration. Researchers continue to refine Newton's method to improve convergence rates and enhance its practical applicability. Multipoint iterative methods have emerged as the most efficient root-solvers, surpassing the theoretical limitations of one-point methods in terms of convergence order and computational efficiency. These advantages have led to a surge in interest in multipoint methods in the current era. The advancement of symbolic computation and multi-precision arithmetic has further accelerated their development.

However, while improving the convergence rate is advantageous, it may also lead to a higher number of function evaluations, which can ultimately decrease the efficiency index of these methods. The efficiency index, as discussed in [2], measures the balance between the order of convergence and the number of functional evaluations per step, represented by the formula $E = \rho^{1/\gamma}$, where ρ is the order of convergence and γ is the number of functional and derivative evaluations conducted per iteration. Researchers have a particular interest in developing accelerated multipoint methods with memory due to their high computational efficiency. They aim to enhance the order of these methods beyond the limits of optimal methods, guided by Kung–Traub's conjecture, which posits that $n+1$ function evaluations can achieve the optimal convergence order of 2^n [2].

However, iterative methods that incorporate memory leverage the state of recent and previous iterations to increase the efficiency index as well as the convergence order. Significant efforts have been made recently in the field to expand without-memory methods to with-memory methods by employing self-accelerating parameters. Liu et al. [3] upgraded a single-step without-memory method with a second order of convergence to a with-memory method using one self-accelerating parameter and achieved a fourth order of convergence. Sharma et al. [4] upgraded an eighth-order without-memory iterative method to a with-memory method using two self-accelerating parameters and attained a tenth order of convergence. Additionally, Thangkhenpau et al. [5] developed a derivative-free without-memory iterative method with an eighth order of convergence and then expanded it to a with-memory method using four self-accelerating parameters, which resulted in an increase in the convergence order from 8 to 15.5156. In recent years, the development of with-memory iterative methods has garnered considerable interest among researchers. Notable contributors to the development of with-memory methods include Choubey et al. [6–8], Raziyeh Erfanifar [9], Howk et al. [10], Sharma et al. [11,12], Wang and Zhang [13], Liu et al. [14], and Panday et al. [15].

In this article, we proposed a new three-step, uni-parametric, with-memory iterative technique while involving complex error computations. It is designed for modular implementation to facilitate practical adoption in solving nonlinear equations efficiently and it improves the R-order of convergence from 8 to 8.8989. An efficiency index of 1.7272 is attained by this method. By adding a self-accelerating parameter to the third step of an existing optimal eighth-order without-memory iterative technique [16] and conducting a comprehensive convergence analysis, a new uni-parametric three-point with-memory iterative method was developed, as described in Section 2. A detailed assessment using numerical tests is presented in Section 3, which offers a thorough comparison of the suggested method with other well-established methods. Section 4 provides a comprehensive summary of the research findings and their implications.

2. Analysis of Convergence for With-Memory Method

In this section, we use a parameter α in the third step of the three-step optimal eighth-order scheme proposed by Matthies et al. [16] in 2016, to increase its order of convergence from eight to nine. After adding the parameter α in the third step of [16], we obtain

$$
\begin{aligned}
y_n &= x_n - \frac{\zeta(x_n)}{\zeta'(x_n)},\\
z_n &= x_n - \frac{\zeta(x_n)}{\zeta'(x_n)}\left(1 + \frac{\zeta(y_n)}{\zeta(x_n)} + \left(1 + \frac{1}{1+\frac{\zeta(x_n)}{\zeta'(x_n)}}\right)\left(\frac{\zeta(y_n)}{\zeta(x_n)}\right)^2\right),\\
x_{n+1} &= z_n - \frac{\zeta(z_n)}{\zeta[z_n,y_n] + (z_n-y_n)\zeta[z_n,y_n,x_n] + (z_n-y_n)(z_n-x_n)\zeta[z_n,y_n,x_n,x_n] + \alpha\zeta(z_n)},
\end{aligned} \tag{2}
$$

where $\zeta[z_n, y_n]$, $\zeta[z_n, y_n, x_n]$, and $\zeta[z_n, y_n, x_n, x_n]$ represent the divided differences and are defined by $\zeta[z_n, y_n] = \frac{\zeta(z_n)-\zeta(y_n)}{z_n-y_n}$, $\zeta[z_n, y_n, x_n] = \frac{\zeta[z_n,y_n]-\zeta[y_n,x_n]}{z_n-x_n}$, and $\zeta[z_n, y_n, x_n, x_n] = \frac{\zeta[z_n,y_n,x_n]-\zeta[y_n,x_n,x_n]}{z_n-x_n}$.

Using Taylor-series approximation, the expressions for $\zeta(x_n)$ and $\zeta'(x_n)$ can be written as

$$\zeta(x_n) = A(e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + c_5e_n^5 + c_6e_n^6 + c_7e_n^7 + c_8e_n^8) + O\left(e_n^9\right), \tag{3}$$

$$\zeta'(x_n) = A(1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + 5c_5e_n^4 + 6c_6e_n^5 + 7c_7e_n^6 + 8c_8e_n^7 + 9c_9e_n^8) + O\left(e_n^9\right), \tag{4}$$

where $A = \zeta'(\xi)$, ξ is the simple root of $\zeta(x)$, $e_n = x_n - \xi$, and $c_j = \frac{\zeta^{(j)}(\xi)}{j!\zeta'(\xi)}$ for $j = 2, 3, \ldots$.

Now, the error expressions for the first two sub-steps of (2) is given by [16]:

$$e_{n,y} = c_2e_n^2 + \left(-2c_2^2 + 2c_3\right)e_n^3 + \left(4c_2^3 - 7c_2c_3 + 3c_4\right)e_n^4 + O\left(e_n^5\right), \tag{5}$$

$$\begin{aligned} e_{n,z} =& c_2\left(c_2 + 5c_2^2 - c_3\right)e_n^4 \\ &+ \left(-8c_2^3 - 36c_2^4 - 2c_3^2 + c_2^2(-1 + 32c_3) + c_2(4c_3 - 2c_4)\right)e_n^5 + O\left(e_n^6\right), \end{aligned} \tag{6}$$

and we obtain the error expansion for the third sub-step of (2) as:

$$\begin{aligned} e_{n+1} =\ & c_2^2\left(c_2 + 5c_2^2 - c_3\right)\left((\alpha + c_2)(c_2 + 5c_2^2 - c_3) + c_4\right)e_n^8 \\ & - c_2\Big(8(19 + 45\alpha)c_2^6 + 360c_2^7 + 2c_2^5(13 + 76\alpha - 196c_3) + 4c_3^2(-\alpha c_3 + c_4) \\ & + c_2^4(2 + 26\alpha - 8(15 + 49\alpha)c_3 + 66c_4) + 2c_2^3(\alpha - 5(1 + 12\alpha)c_3 + 42c_3^2 \\ & + (7 + 10\alpha)c_4 - 5c_5) + c_2^2(2c_3(-5\alpha + 6(1 + 7\alpha)c_3) + c_4 + 4(\alpha - 12c_3)c_4 - 2c_5) \\ & + 2c_2(6\alpha c_3^2 - 2c_3^3 + c_4^2 + c_3(-((3 + 2\alpha)c_4) + c_5))\Big)e_n^9 + O\left(e_n^{10}\right), \end{aligned} \tag{7}$$

where $e_{n,y} = y_n - \xi$, $e_{n,z} = z_n - \xi$, and $\alpha \in R$. We obtain the following with-memory iterative scheme by replacing α with a self-accelerating parameter α_n in (2):

$$\begin{aligned} y_n &= x_n - \frac{\zeta(x_n)}{\zeta'(x_n)}, \\ z_n &= x_n - \frac{\zeta(x_n)}{\zeta'(x_n)}\left(1 + \frac{\zeta(y_n)}{\zeta(x_n)} + \left(1 + \frac{1}{1 + \frac{\zeta(x_n)}{\zeta'(x_n)}}\right)\left(\frac{\zeta(y_n)}{\zeta(x_n)}\right)^2\right), \\ x_{n+1} &= z_n - \frac{\zeta(z_n)}{\zeta[z_n, y_n] + (z_n - y_n)\zeta[z_n, y_n, x_n] + (z_n - y_n)(z_n - x_n)\zeta[z_n, y_n, x_n, x_n] + \alpha_n\zeta(z_n)}. \end{aligned} \tag{8}$$

The above scheme is represented by NWM9. Now, from (7), it is clear that the convergence order of the algorithm (2) is eight when $\alpha \neq -\frac{c_4}{c_2 + 5c_2^2 - c_3} - c_2$. Next, to accelerate the order of convergence of the algorithm presented in (8) from eight to nine, we can assume $\alpha = -\frac{c_4}{c_2 + 5c_2^2 - c_3} - c_2 = -\frac{\zeta^{(iv)}(\xi)\zeta'(\xi)}{12\zeta''(\xi)\zeta'(\xi) + 30(\zeta''(\xi))^2 - 4\zeta'''(\xi)\zeta'(\xi)} - \frac{\zeta''(\xi)}{2\zeta'(\xi)}$, but in reality, the exact values of $\zeta'(\xi), \zeta''(\xi), \zeta'''(\xi)$, and $\zeta^{(iv)}(\xi)$ are not attainable in practice. So, we assume the parameter α as α_n. The parameter α_n can be calculated by using the available data from the current and previous iterations and satisfies the condition $\lim_{n\to\infty} \alpha_n = -\frac{c_4}{c_2 + 5c_2^2 - c_3} - c_2 = -\frac{\zeta^{(iv)}(\xi)\zeta'(\xi)}{12\zeta''(\xi)\zeta'(\xi) + 30(\zeta''(\xi))^2 - 4\zeta'''(\xi)\zeta'(\xi)} - \frac{\zeta''(\xi)}{2\zeta'(\xi)}$ such that the eighth-order asymptotic convergence constant should be zero in the error expression (7). The formula for α_n is as follows:

$$\alpha_n = -\frac{H_7^{(iv)}(z_n)\zeta'(x_n)}{12H_5''(x_n)\zeta'(x_n) + 30(H_5''(x_n))^2 - 4H_6'''(y_n)\zeta'(x_n)} - \frac{H_5''(x_n)}{2\zeta'(x_n)}, \tag{9}$$

where the Hermite interpolating polynomials $H_m(x)$ for $m = 5,6,7$ are given by

$$\begin{aligned} H_7(x) = \; & \zeta(z_n) + (x-z_n)\zeta[z_n,y_n] + (x-z_n)(x-y_n)\zeta[z_n,y_n,x_n] \\ & + (x-z_n)(x-y_n)(x-x_n)\zeta[z_n,y_n,x_n,x_n] + (x-z_n)(x-y_n)(x-x_n)^2 \\ & \zeta[z_n,y_n,x_n,x_n,z_{n-1}] + (x-z_n)(x-y_n)(x-x_n)^2(x-z_{n-1}) \\ & \zeta[z_n,y_n,x_n,x_n,z_{n-1},y_{n-1}] + (x-z_n)(x-y_n)(x-x_n)^2(x-z_{n-1})(x-y_{n-1}) \\ & \zeta[z_n,y_n,x_n,x_n,z_{n-1},y_{n-1},x_{n-1}] + (x-z_n)(x-y_n)(x-x_n)^2(x-z_{n-1}) \\ & (x-y_{n-1})(x-x_{n-1})\zeta[z_n,y_n,x_n,x_n,z_{n-1},y_{n-1},x_{n-1},x_{n-1}], \end{aligned}$$

$$\begin{aligned} H_6(x) = \; & \zeta(y_n) + (x-y_n)\zeta[y_n,x_n] + (x-y_n)(x-x_n)\zeta[y_n,x_n,x_n] \\ & + (x-y_n)(x-x_n)^2\zeta[y_n,x_n,x_n,z_{n-1}] + (x-y_n)(x-x_n)^2(x-z_{n-1}) \\ & \zeta[y_n,x_n,x_n,z_{n-1},y_{n-1}] + (x-y_n)(x-x_n)^2(x-z_{n-1})(x-y_{n-1}) \\ & \zeta[y_n,x_n,x_n,z_{n-1},y_{n-1},x_{n-1}] + (x-y_n)(x-x_n)^2(x-z_{n-1})(x-y_{n-1}) \\ & (x-x_{n-1})\zeta[y_n,x_n,x_n,z_{n-1},y_{n-1},x_{n-1},x_{n-1}], \end{aligned}$$

$$\begin{aligned} H_5(x) = \; & \zeta(x_n) + (x-x_n)\zeta[x_n,x_n] + (x-x_n)^2\zeta[x_n,x_n,z_{n-1}] \\ & + (x-x_n)^2(x-z_{n-1})\zeta[x_n,x_n,z_{n-1},y_{n-1}] + (x-x_n)^2(x-z_{n-1})(x-y_{n-1}) \\ & \zeta[x_n,x_n,z_{n-1},y_{n-1},x_{n-1}] + (x-x_n)^2(x-z_{n-1})(x-y_{n-1})(x-x_{n-1}) \\ & \zeta[x_n,x_n,z_{n-1},y_{n-1},x_{n-1},x_{n-1}], \end{aligned}$$

Note: The condition $H'_m(x_n) = \zeta'(x_n)$ is satisfied by the Hermite interpolation polynomial $H_m(x)$ for $m = 5,6,7$. So, $\alpha_n = -\frac{H_7^{(iv)}(z_n)\zeta'(x_n)}{12H''_5(x_n)\zeta'(x_n)+30(H''_5(x_n))^2-4H'''_6(y_n)\zeta'(x_n)} - \frac{H''_5(x_n)}{2\zeta'(x_n)}$ can be expressed as $\alpha_n = -\frac{H_7^{(iv)}(z_n)H'_m(x_n)}{12H''_5(x_n)H'_m(x_n)+30(H''_5(x_n))^2-4H'''_6(y_n)H'_m(x_n)} - \frac{H''_5(x_n)}{2H'_m(x_n)}$ for $m = 5,6,7$.

Theorem 1. *Let H_m be the Hermite polynomial of degree m, interpolating the function ζ at interpolation nodes $z_n, y_n, x_n, x_n, z_{n-1}, y_{n-1}, x_{n-1}, x_{n-1}$ within an interval $D \subset \mathbb{R}$ and the derivative $\zeta^{(m+1)}$ is continuous in D with $H_m(x_n) = \zeta(x_n)$ and $H'_m(x_n) = \zeta'(x_n)$. Suppose that all nodes $z_n, y_n, x_n, x_n, z_{n-1}, y_{n-1}, x_{n-1}, x_{n-1}$ are in the neighborhood of the root ξ. Then,*

$$H_7^{(iv)}(z_n) = 24\zeta'(\xi)(c_4 - c_8 e_{n-1,z}e_{n-1,y}e_{n-1}^2), \tag{10}$$

$$H'''_6(y_n) = 6\zeta'(\xi)(c_3 - c_7 e_{n-1,z}e_{n-1,y}e_{n-1}^2), \tag{11}$$

$$H''_5(x_n) = 2\zeta'(\xi)(c_2 - c_6 e_{n-1,z}e_{n-1,y}e_{n-1}^2), \tag{12}$$

and

$$\begin{aligned} \alpha_n = & -\frac{H_7^{(iv)}(z_n)\zeta'(x_n)}{12H''_5(x_n)\zeta'(x_n)+30(H''_5(x_n))^2-4H'''_6(y_n)\zeta'(x_n)} - \frac{H''_5(x_n)}{2\zeta'(x_n)} \\ & \sim -\frac{c_4}{c_2+5c_2^2-c_3} - c_2 + \frac{c_2}{c_2+5c_2^2-c_3}e_{n-1,z}e_{n-1,y}e_{n-1}^2 - \frac{c_6c_8}{c_2+5c_2^2-c_3}e_{n-1,z}^2e_{n-1,y}^2e_{n-1}^4 \\ & + \frac{5c_2^2c_8}{c_2+5c_2^2-c_3}e_{n-1,z}e_{n-1,y}e_{n-1}^2 + \frac{5c_6^2c_8}{c_2+5c_2^2-c_3}e_{n-1,z}^3e_{n-1,y}^3e_{n-1}^6 \\ & - \frac{10c_2c_6c_8}{c_2+5c_2^2-c_3}e_{n-1,z}e_{n-1,y}e_{n-1}^2 - \frac{c_3c_8}{c_2+5c_2^2-c_3}e_{n-1,z}e_{n-1,y}e_{n-1}^2 \\ & + \frac{c_7c_8}{c_2+5c_2^2-c_3}e_{n-1,z}^2e_{n-1,y}^2e_{n-1}^4. \end{aligned} \tag{13}$$

Again, after simplification, we have

$$\alpha_n + \frac{c_4}{c_2 + 5c_2^2 - c_3} + c_2 = (\alpha_n + c_2)(c_2 + 5c_2^2 - c_3) + c_4$$
$$\sim \Big(\frac{c_2}{c_2 + 5c_2^2 - c_3} - \frac{c_6 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z} e_{n-1,y} e_{n-1}^2 + \frac{5c_2^2 c_8}{c_2 + 5c_2^2 - c_3}$$
$$+ \frac{5c_6^2 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z}^2 e_{n-1,y}^2 e_{n-1}^4 - \frac{10c_2 c_6 c_8}{c_2 + 5c_2^2 - c_3} - \frac{c_3 c_8}{c_2 + 5c_2^2 - c_3}$$
$$+ \frac{c_7 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z} e_{n-1,y} e_{n-1}^2 \Big) e_{n-1,z} e_{n-1,y} e_{n-1}^2. \tag{14}$$

Proof. We can calculate the expression of the seventh-degree, sixth-degree, and fifth-degree Hermite interpolation polynomial as

$$\zeta(x) - H_7(x) = \frac{\zeta^{(8)}(\delta)}{8!}(x - z_n)(x - y_n)(x - x_n)^2 (x - z_{n-1})(x - y_{n-1})(x - x_{n-1})^2, \tag{15}$$

$$\zeta(x) - H_6(x) = \frac{\zeta^{(7)}(\delta)}{7!}(x - y_n)(x - x_n)^2 (x - z_{n-1})(x - y_{n-1})(x - x_{n-1})^2, \tag{16}$$

$$\zeta(x) - H_5(x) = \frac{\zeta^{(6)}(\delta)}{6!}(x - x_n)^2 (x - z_{n-1})(x - y_{n-1})(x - x_{n-1})^2. \tag{17}$$

Now, we obtain the below-mentioned equations by rearranging and differentiating Equation (15) four times at the point $x = z_n$, Equation (16) three times at the point $x = y_n$, and Equation (17) two times at the point $x = x_n$, respectively.

$$H_7^{(iv)}(z_n) = \zeta^{(iv)}(z_n) - 24\frac{\zeta^{(8)}(\delta)}{8!}(z_n - z_{n-1})(z_n - y_{n-1})(z_n - x_{n-1})^2, \tag{18}$$

$$H_6'''(y_n) = \zeta'''(y_n) - 6\frac{\zeta^{(7)}(\delta)}{7!}(y_n - z_{n-1})(y_n - y_{n-1})(y_n - x_{n-1})^2, \tag{19}$$

$$H_5''(x_n) = \zeta''(x_n) - 2\frac{\zeta^{(6)}(\delta)}{6!}(x_n - z_{n-1})(x_n - y_{n-1})(x_n - x_{n-1})^2. \tag{20}$$

Next, Taylor's series expansion of ζ' at the points z_n, y_n, and x_n in D and $\delta \in D$ about the zero ξ of ζ provides

$$\zeta'(x_n) = \zeta'(\xi)\Big(1 + 2c_2 e_n + 3c_3 e_n^2 + O(e_n^3)\Big), \tag{21}$$

$$\zeta''(x_n) = \zeta'(\xi)\Big(2c_2 + 6c_3 e_n + O(e_n^2)\Big), \tag{22}$$

$$\zeta'''(y_n) = \zeta'(\xi)\Big(6c_3 + 24c_4 e_{n,y} + O(e_{n,y}^2)\Big). \tag{23}$$

Similarly,

$$\zeta^{(iv)}(z_n) = \zeta'(\xi)\Big(24c_4 + 120c_5 e_{n,z} + O(e_{n,z}^2)\Big), \tag{24}$$

$$\zeta^{(6)}(\delta) = \zeta'(\xi)\Big(6!c_6 + 7!c_7 e_\delta + O(e_\delta^2)\Big), \tag{25}$$

$$\zeta^{(7)}(\delta) = \zeta'(\xi)\Big(7!c_7 + 8!c_8 e_\delta + O(e_\delta^2)\Big), \tag{26}$$

$$\zeta^{(8)}(\delta) = \zeta'(\xi)\Big(8!c_8 + 9!c_9 e_\delta + O(e_\delta^2)\Big), \tag{27}$$

where $e_\delta = \delta - \xi$. Putting (24) and (27) in (18), (23) and (26) in (19), and (22) and (25) in (20), we obtain

$$H_7^{(iv)}(z_n) = 24\zeta'(\xi)(c_4 - c_8 e_{n-1,z} e_{n-1,y} e_{n-1}^2), \tag{28}$$

$$H_6'''(y_n) = 6\zeta'(\xi)(c_3 - c_7 e_{n-1,z} e_{n-1,y} e_{n-1}^2), \tag{29}$$

and

$$H_5''(x_n) = 2\zeta'(\xi)(c_2 - c_6 e_{n-1,z} e_{n-1,y} e_{n-1}^2), \tag{30}$$

Using Equations (21), (28), (29) and (30), we have

$$\begin{aligned}
&-\frac{H_7^{(iv)}(z_n)\zeta'(x_n)}{12H_5''(x_n)\zeta'(x_n) + 30(H_5''(x_n))^2 - 4H_6'''(y_n)\zeta'(x_n)} - \frac{H_5''(x_n)}{2\zeta'(x_n)} \\
&\sim -\frac{c_4}{c_2 + 5c_2^2 - c_3} - c_2 + \frac{c_2}{c_2 + 5c_2^2 - c_3} e_{n-1,z} e_{n-1,y} e_{n-1}^2 - \frac{c_6 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z}^2 e_{n-1,y}^2 e_{n-1}^4 \\
&+ \frac{5c_2^2 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z} e_{n-1,y} e_{n-1}^2 + \frac{5c_6^2 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z}^3 e_{n-1,y}^3 e_{n-1}^6 \\
&- \frac{10c_2 c_6 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z} e_{n-1,y} e_{n-1}^2 - \frac{c_3 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z} e_{n-1,y} e_{n-1}^2 \\
&+ \frac{c_7 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z}^2 e_{n-1,y}^2 e_{n-1}^4.
\end{aligned} \tag{31}$$

And hence,

$$\begin{aligned}
\alpha_n \sim &-\frac{c_4}{c_2 + 5c_2^2 - c_3} - c_2 + \Bigg(\frac{c_2}{c_2 + 5c_2^2 - c_3} - \frac{c_6 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z} e_{n-1,y} e_{n-1}^2 \\
&+ \frac{5c_2^2 c_8}{c_2 + 5c_2^2 - c_3} + \frac{5c_6^2 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z}^2 e_{n-1,y}^2 e_{n-1}^4 - \frac{10c_2 c_6 c_8}{c_2 + 5c_2^2 - c_3} - \frac{c_3 c_8}{c_2 + 5c_2^2 - c_3} \\
&+ \frac{c_7 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z} e_{n-1,y} e_{n-1}^2\Bigg) e_{n-1,z} e_{n-1,y} e_{n-1}^2.
\end{aligned} \tag{32}$$

or

$$\begin{aligned}
\alpha_n + \frac{c_4}{c_2 + 5c_2^2 - c_3} + c_2 &= (\alpha_n + c_2)(c_2 + 5c_2^2 - c_3) + c_4 \\
&\sim \Bigg(\frac{c_2}{c_2 + 5c_2^2 - c_3} - \frac{c_6 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z} e_{n-1,y} e_{n-1}^2 + \frac{5c_2^2 c_8}{c_2 + 5c_2^2 - c_3} \\
&+ \frac{5c_6^2 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z}^2 e_{n-1,y}^2 e_{n-1}^4 - \frac{10c_2 c_6 c_8}{c_2 + 5c_2^2 - c_3} - \frac{c_3 c_8}{c_2 + 5c_2^2 - c_3} \\
&+ \frac{c_7 c_8}{c_2 + 5c_2^2 - c_3} e_{n-1,z} e_{n-1,y} e_{n-1}^2\Bigg) e_{n-1,z} e_{n-1,y} e_{n-1}^2.
\end{aligned} \tag{33}$$

This completes the proof of Theorem 1. □

The definition of R-order of convergence [17] and the statement in [18] can be used to estimate the order of convergence of the iterative scheme (8).

Theorem 2. *If the errors $e_j = x_j - \xi$ evaluated by an iterative root-finding method IM fulfill*

$$e_{k+1} \sim \prod_{i=0}^{m-2} (e_{k-i})^{m_i}, k \geq k(\{e_k\}) \tag{34}$$

then the R-order of convergence of IM, denoted with $O_R(IM, \xi)$, satisfies the inequality $O_R(IM, \xi) \geq s^*$, where s^* is the unique positive solution of the equation $s_{n+1} - \sum_{i=0}^{n} m_i s^{n-i} = 0$ [18].

Proof. The proof of the above Theorem 2 can be found in [18]. □

Presently, for the new iterative scheme with memory (8), we can state the subsequent convergence theorem.

Theorem 3. *In the iterative method (8), let α_n be a varying parameter, calculated by Equation (9). If an initial guess x_0 is sufficiently near to a simple zero ξ of $\zeta(x)$, then the R-order of convergence of the iterative method (8) with memory is at least 8.8989.*

Proof. Let the iterative method (IM) generate the sequence of $\{x_n\}$ which converges to the root ξ of $\zeta(x)$. By means of R-order $O_R(IM, \xi) \geq r$, we express

$$e_{n+1} \sim D_{n,r} e_n^r, \tag{35}$$

and

$$e_n \sim D_{n-1,r} e_{n-1}^r. \tag{36}$$

Next, $D_{n,r}$ will tend to the asymptotic error constant D_r of IM by taking $n \to \infty$; then,

$$e_{n+1} \sim D_{n,r}(D_{n-1,r} e_{n-1}^r)^r = D_{n,r} D_{n-1,r}^r e_{n-1}^{r^2}. \tag{37}$$

The resulting error expression of the with-memory scheme (8) can be obtained using (5)–(7) and the varying parameter α_n.

$$e_{n,y} = y_n - \xi \sim c_2 e_n^2, \tag{38}$$

$$e_{n,z} = z_n - \xi \sim c_2(c_2 + 5c_2^2 - c_3) e_n^4, \tag{39}$$

and

$$e_{n+1} = x_{n+1} - \xi \sim c_2^2 \Big(c_2 + 5c_2^2 - c_3\Big) \Big((\alpha + c_2)(c_2 + 5c_2^2 - c_3) + c_4\Big) e_n^8. \tag{40}$$

Here, the higher-order terms in Equations (38)–(40) are excluded.

Now, let the R-order convergence of the iterative sequences $\{y_n\}$ and $\{z_n\}$ be p and q, respectively; then,

$$e_{n,y} \sim D_{n,p} e_n^p \sim D_{n,p}(D_{n-1,r} e_{n-1}^r)^p = D_{n,p} D_{n-1,r}^p e_{n-1}^{rp}, \tag{41}$$

and

$$e_{n,z} \sim D_{n,q} e_n^q \sim D_{n,q}(D_{n-1,r} e_{n-1}^r)^q = D_{n,q} D_{n-1,r}^q e_{n-1}^{rq}. \tag{42}$$

Now, by Equations (36) and (38), we obtain

$$e_{n,y} \sim c_2 e_n^2 \sim c_2 (D_{n-1,r} e_{n-1}^r)^2 \sim c_2 D_{n-1,r}^2 e_{n-1}^{2r}. \tag{43}$$

Also, by Equations (36) and (39), we obtain

$$\begin{aligned} e_{n,z} &\sim c_2(c_2 + 5c_2^2 - c_3) e_n^4 \\ &\sim c_2(c_2 + 5c_2^2 - c_3)(D_{n-1,r} e_{n-1}^r)^4 \\ &\sim c_2(c_2 + 5c_2^2 - c_3) D_{n-1,r}^4 e_{n-1}^{4r}. \end{aligned} \tag{44}$$

Again, by Equations (33), (36) and (40), we have

$$
\begin{aligned}
e_{n+1} &\sim c_2^2\Big(c_2+5c_2^2-c_3\Big)\Big((\alpha+c_2)(c_2+5c_2^2-c_3)+c_4\Big)e_n^8\\
&\sim c_2^2\Big(c_2+5c_2^2-c_3\Big)\Big(\frac{c_2}{c_2+5c_2^2-c_3}-\frac{c_6c_8}{c_2+5c_2^2-c_3}e_{n-1,z}e_{n-1,y}e_{n-1}^2+\frac{5c_2^2c_8}{c_2+5c_2^2-c_3}\\
&+\frac{5c_6^2c_8}{c_2+5c_2^2-c_3}e_{n-1,z}^2e_{n-1,y}^2e_{n-1}^4-\frac{10c_2c_6c_8}{c_2+5c_2^2-c_3}-\frac{c_3c_8}{c_2+5c_2^2-c_3}\\
&+\frac{c_7c_8}{c_2+5c_2^2-c_3}e_{n-1,z}e_{n-1,y}e_{n-1}^2\Big)e_{n-1,z}e_{n-1,y}e_{n-1}^2(D_{n-1,r}e_{n-1}^r)^8\\
&\sim c_2^2\Big(c_2+5c_2^2-c_3\Big)\Big(\frac{c_2}{c_2+5c_2^2-c_3}-\frac{c_6c_8}{c_2+5c_2^2-c_3}e_{n-1,z}e_{n-1,y}e_{n-1}^2+\frac{5c_2^2c_8}{c_2+5c_2^2-c_3}\\
&+\frac{5c_6^2c_8}{c_2+5c_2^2-c_3}e_{n-1,z}^2e_{n-1,y}^2e_{n-1}^4-\frac{10c_2c_6c_8}{c_2+5c_2^2-c_3}-\frac{c_3c_8}{c_2+5c_2^2-c_3}\\
&+\frac{c_7c_8}{c_2+5c_2^2-c_3}e_{n-1,z}e_{n-1,y}e_{n-1}^2\Big)D_{n-1,q}e_{n-1}^qD_{n-1,p}e_{n-1}^pe_{n-1}^2(D_{n-1,r}e_{n-1}^r)^8\\
&\sim c_2^2\Big(c_2+5c_2^2-c_3\Big)\Big(\frac{c_2}{c_2+5c_2^2-c_3}-\frac{c_6c_8}{c_2+5c_2^2-c_3}e_{n-1,z}e_{n-1,y}e_{n-1}^2+\frac{5c_2^2c_8}{c_2+5c_2^2-c_3}\\
&+\frac{5c_6^2c_8}{c_2+5c_2^2-c_3}e_{n-1,z}^2e_{n-1,y}^2e_{n-1}^4-\frac{10c_2c_6c_8}{c_2+5c_2^2-c_3}-\frac{c_3c_8}{c_2+5c_2^2-c_3}\\
&+\frac{c_7c_8}{c_2+5c_2^2-c_3}e_{n-1,z}e_{n-1,y}e_{n-1}^2\Big)D_{n-1,q}D_{n-1,p}D_{n-1,r}^8e_{n-1}^{8r+p+q+2}.
\end{aligned}
\tag{45}
$$

since $r > q > p$. By equating the exponents of e_{n-1} present in the set of relations (41)–(43), (42)–(44), and (37)–(45), we attain the resulting system of equations:

$$
\begin{aligned}
rp &= 2r\\
rq &= 4r\\
r^2 &= 8r+p+q+2
\end{aligned}
\tag{46}
$$

The solution of the system of Equation (46) is specified by $r = 8.8989$, $q = 4$, and $p = 2$. As a result, the R-order of convergence of the with-memory iterative method (8) is at least 8.8989. □

3. Numerical Discussion

This section examines the convergence behavior of the newly developed with-memory technique NWM9 introduced in (8). Our goal is to evaluate the effectiveness of a recently developed iterative method by applying it to a variety of nonlinear problems. The nonlinear test functions, along with their roots and initial guesses for our numerical analysis, are described below:

Example 1: $\zeta_1(x) = 1 + x^2 \cdot e^{\cos x/2} - (x+1)e^{\sin x/2}$, $x_0 = 0.9$, $\xi \approx 0.8475$
Example 2: $\zeta_2(x) = e^{x^3+\cos x+1} - x^2 + x + 1$, $x_0 = -0.8$, $\xi \approx -1.0787$
Example 3: $\zeta_3(x) = x \cdot e^{x^2} - \sin^2 x + 3\cos x + 5$, $x_0 = -1.2$, $\xi \approx -1.2076$
Example 4: $\zeta_4(x) = e^{-x^2}(1 + x^3 + x^6)(x-2)$, $x_0 = 1.95$, $\xi \approx 2.0000$
Example 5: $\zeta_5(x) = x^7 - 4x^4 + x - 1$, $x_0 = 1.58$, $\xi \approx 1.5749$
Example 6: $\zeta_6(x) = e^{x^2-4} + \sin(x-2) - x^4 + 15$, $x_0 = 2.1$, $\xi \approx 2.0000$
Example 7: $\zeta_7(x) = e^{-x^2+x+2} - 1$, $x_0 = 2.01$, $\xi \approx 2.0000$
Example 8 [19,20]: In civil engineering, beams in mathematical models are horizontal elements that support loads and span openings, sometimes called lintels if made of stone or brick. "Floor joist" or "roof joist" designates beams supporting floors or roofs, respectively. Stringers support lighter bridge deck loads, while floor beams handle heavier transverse

loads. Girders, constructed from metal plates or concrete, bear terminal loads of smaller beams, enhancing rigidity and extending spans. Various nonlinear mathematical models have been developed to specify the precise beam location. The model below is an example which was taken from [19,20]:

$$\zeta_8(x) = x^4 + 4x^3 - 24x^2 + 16x + 16 = 0. \tag{47}$$

The roots of the above fourth-order polynomial are 2, 2 and $-4 \pm 2\sqrt{3}$ and the initial guess for $\zeta_8(x)$ is taken as $x_0 = -0.55''$.

We compare our method NWM9 (8) to various well-established methods published in the literature, including MSSV8 (48), ACD8 (49), LE8 (50), SH8 (51), BAC8 (52), and TKM9 (53), which are discussed below:

In 2016, Matthies et al. (MSSV8) [16] developed an optimal eighth-order iterative method which is defined as

$$\begin{aligned}
y_n &= x_n - \frac{\zeta(x_n)}{\zeta'(x_n)},\\
z_n &= x_n - \frac{\zeta(x_n)}{\zeta'(x_n)}\left(1 + \frac{\zeta(y_n)}{\zeta(x_n)} + \left(1 + \frac{1}{1 + \frac{\zeta(x_n)}{\zeta'(x_n)}}\right)\left(\frac{\zeta(y_n)}{\zeta(x_n)}\right)^2\right),\\
x_{n+1} &= z_n - \frac{\zeta(z_n)}{\zeta[z_n, y_n] + (z_n - y_n)\zeta[z_n, y_n, x_n] + (z_n - y_n)(z_n - x_n)\zeta[z_n, y_n, x_n, x_n]}.
\end{aligned} \tag{48}$$

In 2024, Abdullah et al. (ACD8) [21] developed an optimal eighth-order iterative method which is defined as

$$\begin{aligned}
y_n &= x_n - \frac{\zeta(x_n)}{\zeta'(x_n)},\\
z_n &= x_n - \frac{\zeta(x_n)(\zeta(x_n)^3 + 2\zeta(x_n)\zeta(y_n)^2)}{\zeta'(x_n)(\zeta(x_n) - \zeta(y_n))(\zeta(x_n)^2 + \zeta(y_n)^2)},\\
x_{n+1} &= z_n - \frac{\zeta(z_n)(z_n - y_n)}{\zeta(z_n) - \zeta(y_n)}(A(u_n) + B(v_n) + H(w_n)),
\end{aligned} \tag{49}$$

where $u_n = \frac{\zeta(z_n)}{\zeta(y_n)}$, $v_n = \frac{\zeta(z_n)}{\zeta(x_n)}$, and $w_n = \frac{\zeta(y_n)}{\zeta(x_n)}$.

In 2014, Lotfi and Eftekhari (LE8) [22] developed an optimal eighth-order iterative method which is defined as

$$\begin{aligned}
y_n &= x_n - \frac{\zeta(x_n)}{\zeta'(x_n)},\\
z_n &= y_n - \frac{\zeta(x_n)}{\zeta(x_n) - 2\zeta(y_n)}\frac{\zeta(y_n)}{\zeta'(x_n)},\\
x_{n+1} &= z_n - (K(t_1) \times L(t_2) \times P(t_3))\frac{\zeta(z_n)\zeta[x_n, y_n]}{\zeta[x_n, z_n]\zeta[y_n, z_n]},
\end{aligned} \tag{50}$$

where $t_1 = \frac{\zeta(z_n)}{\zeta(x_n)}$, $t_2 = \frac{\zeta(y_n)}{\zeta(x_n)}$, and $t_3 = \frac{\zeta(z_n)}{\zeta(y_n)}$.

In 2020, Solaiman and Hashim (SH8) [23] developed an optimal eighth-order iterative method which is defined as

$$
\begin{aligned}
y_n &= x_n - \frac{\zeta(x_n)}{\zeta'(x_n)}, \\
z_n &= y_n - \frac{\zeta(y_n)}{\zeta'(y_n)} - \frac{2(\zeta(y_n))^2\zeta'(y_n)R(x_n,y_n)}{4(\zeta'(y_n))^4 - 4\zeta(y_n)(\zeta'(y_n))^2R(x_n,y_n) + (\zeta'(y_n))^2(R(x_n,y_n))^2}, \\
x_{n+1} &= z_n - \frac{\zeta(z_n)}{\zeta'(z_n)},
\end{aligned}
\tag{51}
$$

where $\zeta'(y_n)$, $\zeta'(z_n)$, and $R(x_n, y_n)$ are approximated as

$$
\begin{aligned}
&\zeta'(y_n) \approx 2\zeta[y_n, x_n] - \zeta'(x_n). \\
&\zeta'(z_n) \approx \zeta[z_n, x_n](2 + \tfrac{x_n - z_n}{y_n - z_n}) - \tfrac{(x_n - z_n)^2}{(x_n - y_n)(y_n - z_n)}\zeta[x_n, y_n] + \zeta'(x_n)\tfrac{y_n - z_n}{x_n - y_n}. \\
&R(x_n, y_n) \approx \left[3\tfrac{\zeta(y_n) - \zeta(x_n)}{y_n - x_n} - 2\zeta'(y_n) - \zeta'(x_n)\right]\tfrac{2}{x_n - y_n}.
\end{aligned}
$$

In 2020, Behl et al. (BAC8) [24] developed an optimal eighth-order iterative method which is defined as

$$
\begin{aligned}
w_n &= x_n + \beta\zeta(x_n); \beta \in R, \\
y_n &= x_n - \frac{\zeta(x_n)}{\zeta[w_n, x_n]}, \\
z_n &= y_n - \frac{\zeta(w_n)\zeta(y_n)(y_n - x_n)}{(\zeta(w_n) - \zeta(y_n))(\zeta(y_n) - \zeta(x_n))}, \\
x_{n+1} &= z_n - \frac{\zeta(z_n)(w_n - x_n)(w_n - y_n)(x_n - y_n)}{\zeta[y_n, z_n](w_n - x_n)(w_n - z_n)(x_n - z_n) - a(y_n - z_n)},
\end{aligned}
\tag{52}
$$

where $a = \zeta[x_n, z_n](w_n - y_n)(w_n - z_n) - \zeta[w_n, z_n](x_n - y_n)(x_n - z_n)$.

In 2021, Torkashvand et al. (TKM9) [25] proposed a family of with-memory iterative methods with a ninth order of convergence which is defined as

$$
\begin{aligned}
w_n &= x_n + \beta_n\zeta(x_n), \\
y_n &= x_n - \frac{\zeta(x_n)}{\zeta[x_n, w_n]}, \\
z_n &= y_n - \frac{\zeta[x_n, y_n] - \zeta[y_n, w_n] + \zeta[x_n, w_n]}{\zeta[x_n, y_n]^2}\zeta(y_n), \\
x_{n+1} &= z_n - \frac{\zeta(z_n)}{\zeta[x_n, z_n] + (\zeta[w_n, x_n, y_n] - \zeta[w_n, x_n, z_n] - \zeta[y_n, x_n, z_n])(x_n - z_n)},
\end{aligned}
\tag{53}
$$

where β_n is a self-accelerating parameter and can be calculated as $\beta_n = -\frac{1}{\zeta'(\xi)}$ and approximated by equation (3.11) of [25] with $\beta_0 = 0.001$.

Tables 1–8 summarize the comparative results of all the methods. In these tables, we present the absolute differences between the last two consecutive iterations ($|x_n - x_{n-1}|$) and the absolute residual error ($|\zeta(x_n)|$) of up to three iterations for each function, along with the *COC* for the proposed methods in comparison to some well-known existing methods. The following equation is used to determine the *COC* [26]:

$$
COC = \frac{log|\zeta(x_n)/\zeta(x_{n-1})|}{log|\zeta(x_{n-1})/\zeta(x_{n-2})|}
\tag{54}
$$

All numerical computations were performed using the programming software Mathematica 12.2. To begin the initial iteration of our newly proposed with-memory method NWM9, we set the parameter value α_0 to 0.01.

Table 1. Comparisons of without-memory and with-memory methods after first three (n = 3) iterations for $\zeta_1(x)$.

Method	$\lvert(x_1 - x_0)\rvert$	$\lvert(x_2 - x_1)\rvert$	$\lvert(x_3 - x_2)\rvert$	$\lvert\zeta_1(x_3)\rvert$	COC
MSSV8	0.05252	3.7900×10^{-11}	7.6785×10^{-84}	2.3346×10^{-665}	8.0000
ACD8	0.05252	3.4318×10^{-11}	2.1348×10^{-84}	5.1265×10^{-670}	8.0000
LE8	0.05252	3.9805×10^{-12}	3.7451×10^{-95}	2.4628×10^{-759}	8.0000
SH8	0.05252	4.5810×10^{-12}	2.2774×10^{-92}	9.1021×10^{-735}	8.0000
BAC8	0.05252	1.4930×10^{-10}	1.8619×10^{-78}	1.1669×10^{-621}	8.0000
TKM9	0.05252	1.1405×10^{-11}	3.6015×10^{-96}	7.2790×10^{-817}	8.5292
NWM9	0.05784	4.1302×10^{-11}	1.9899×10^{-93}	3.8586×10^{-831}	8.9618

Table 2. Comparisons of without-memory and with-memory methods after first three (n = 3) iterations for $\zeta_2(x)$.

Method	$\lvert(x_1 - x_0)\rvert$	$\lvert(x_2 - x_1)\rvert$	$\lvert(x_3 - x_2)\rvert$	$\lvert\zeta_2(x_3)\rvert$	COC
MSSV8	0.27875	1.4921×10^{-7}	8.9745×10^{-55}	1.3200×10^{-431}	8.0000
ACD8	0.27875	2.5147×10^{-6}	1.7019×10^{-45}	6.4324×10^{-358}	8.0000
LE8	0.27875	4.6097×10^{-6}	4.9934×10^{-44}	8.1313×10^{-347}	8.0000
SH8	0.27875	1.7169×10^{-6}	2.8891×10^{-48}	1.5956×10^{-381}	8.0000
BAC8	0.26044	1.8314×10^{-2}	1.8518×10^{-11}	1.7828×10^{-82}	7.9934
TKM9	0.27875	1.5174×10^{-6}	1.6804×10^{-51}	5.9549×10^{-434}	8.5281
NWM9	0.27875	7.3581×10^{-8}	7.2640×10^{-62}	1.5269×10^{-544}	8.9548

Table 3. Comparisons of without-memory and with-memory methods after first three (n = 3) iterations for $\zeta_3(x)$.

Method	$\lvert(x_1 - x_0)\rvert$	$\lvert(x_2 - x_1)\rvert$	$\lvert(x_3 - x_2)\rvert$	$\lvert\zeta_3(x_3)\rvert$	COC
MSSV8	0.00765	2.9645×10^{-15}	1.4589×10^{-114}	1.0193×10^{-907}	8.0000
ACD8	0.00765	1.3767×10^{-15}	1.3565×10^{-117}	2.4483×10^{-932}	8.0000
LE8	0.00765	2.1364×10^{-17}	7.5667×10^{-134}	3.8042×10^{-1064}	8.0000
SH8	0.00765	3.8290×10^{-17}	1.5719×10^{-131}	2.5750×10^{-1045}	8.0000
BAC8	0.00765	8.0553×10^{-11}	1.2473×10^{-74}	8.3702×10^{-584}	8.0000
TKM9	0.00765	4.2164×10^{-16}	3.1405×10^{-130}	9.7191×10^{-1103}	8.5327
NWM9	0.00765	2.9645×10^{-15}	4.6276×10^{-129}	6.0799×10^{-1139}	8.8852

Table 4. Comparisons of without-memory and with-memory methods after first three (n = 3) iterations for $\zeta_4(x)$.

Method	$\lvert(x_1 - x_0)\rvert$	$\lvert(x_2 - x_1)\rvert$	$\lvert(x_3 - x_2)\rvert$	$\lvert\zeta_4(x_3)\rvert$	COC
MSSV8	0.05000	7.8095×10^{-10}	9.1958×10^{-72}	4.5442×10^{-567}	8.0000
ACD8	0.05000	7.5624×10^{-10}	1.8286×10^{-72}	2.8568×10^{-573}	8.0000
LE8	0.05000	1.0812×10^{-10}	9.4320×10^{-80}	4.2313×10^{-632}	8.0000
SH8	0.05000	1.6034×10^{-11}	3.3188×10^{-87}	1.4946×10^{-692}	8.0000
BAC8	0.05000	4.5664×10^{-9}	8.6966×10^{-65}	2.0124×10^{-510}	8.0000
TKM9	0.05000	2.4915×10^{-10}	4.9782×10^{-82}	1.3669×10^{-693}	8.5313
NWM9	0.05000	7.7203×10^{-10}	3.6473×10^{-80}	5.5446×10^{-706}	8.9007

Table 5. Comparisons of without-memory and with-memory methods after first three (n = 3) iterations for $\zeta_5(x)$.

Method	$\|(x_1 - x_0)\|$	$\|(x_2 - x_1)\|$	$\|(x_3 - x_2)\|$	$\|\zeta_5(x_3)\|$	*COC*
MSSV8	0.00506	2.8919×10^{-14}	3.9009×10^{-104}	1.9376×10^{-821}	8.0000
ACD8	0.00506	3.5871×10^{-15}	3.6316×10^{-112}	1.8167×10^{-886}	8.0000
LE8	0.00506	1.3336×10^{-16}	1.6545×10^{-125}	4.2086×10^{-995}	8.0000
SH8	0.00506	1.8393×10^{-16}	5.7646×10^{-124}	2.4321×10^{-982}	8.0000
BAC8	0.00506	2.5941×10^{-9}	8.1745×10^{-59}	3.6057×10^{-453}	8.0000
TKM9	0.00506	3.8227×10^{-15}	1.0436×10^{-119}	3.1705×10^{-1010}	8.5323
NWM9	0.00506	2.9141×10^{-14}	3.1787×10^{-117}	7.0560×10^{-1033}	8.9092

Table 6. Comparisons of without-memory and with-memory methods after first three (n = 3) iterations for $\zeta_6(x)$.

Method	$\|(x_1 - x_0)\|$	$\|(x_2 - x_1)\|$	$\|(x_3 - x_2)\|$	$\|\zeta_6(x_3)\|$	*COC*
MSSV8	0.10000	2.1291×10^{-10}	1.9115×10^{-78}	2.1784×10^{-621}	8.0000
ACD8	0.10000	2.7671×10^{-10}	1.7064×10^{-78}	9.6367×10^{-623}	8.0000
LE8	0.10000	7.6557×10^{-10}	6.7672×10^{-75}	6.8097×10^{-594}	8.0000
SH8	0.10000	4.9364×10^{-10}	1.4124×10^{-76}	1.7126×10^{-607}	8.0000
BAC8	0.05633	4.3659×10^{-2}	9.7679×10^{-6}	5.3083×10^{-35}	8.3857
TKM9	0.10000	5.6142×10^{-10}	9.0560×10^{-82}	1.0615×10^{-692}	8.5296
NWM9	0.10000	2.6493×10^{-10}	2.3101×10^{-85}	9.8138×10^{-753}	8.9103

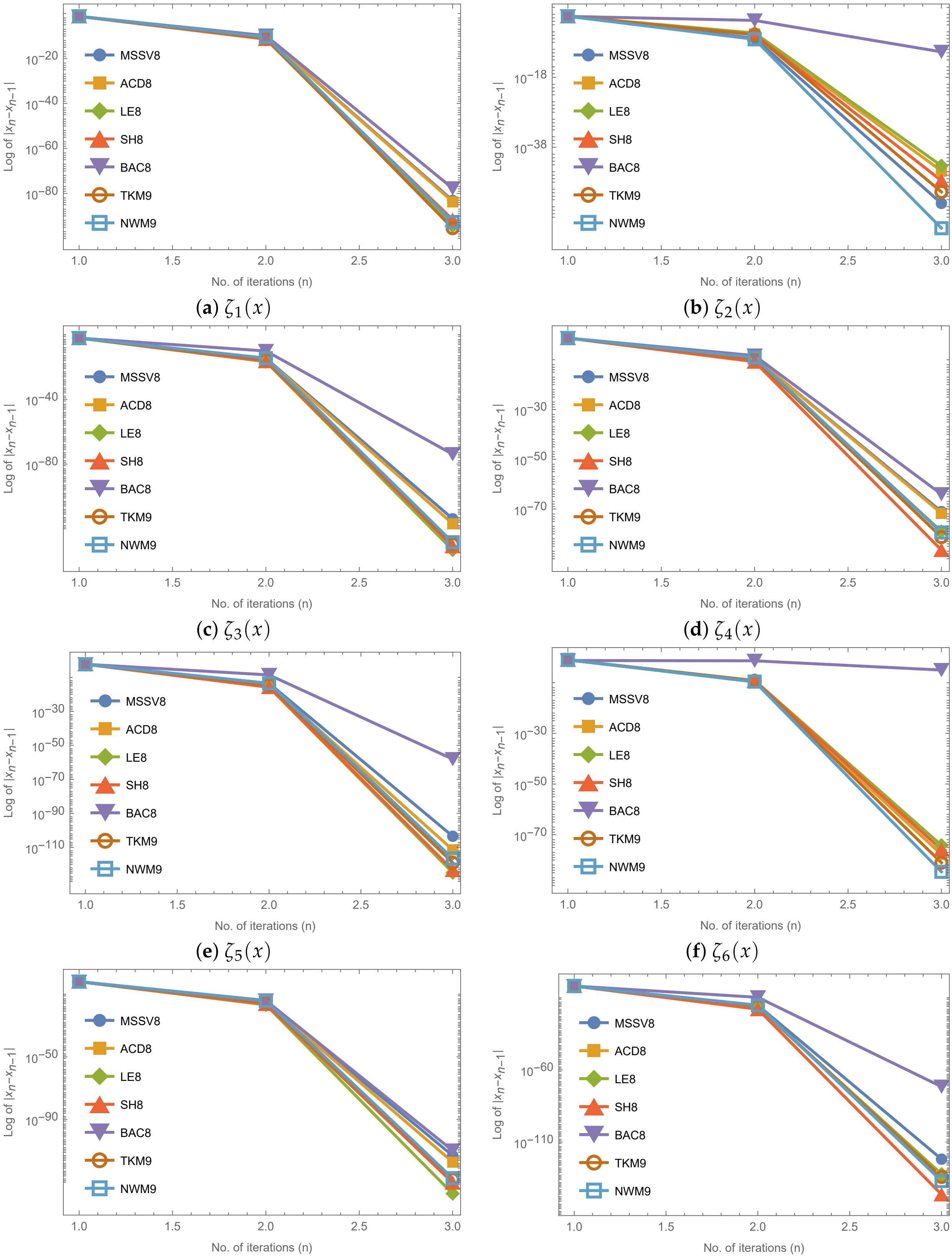
Table 7. Comparisons of without-memory and with-memory methods after first three (n = 3) iterations for $\zeta_7(x)$.

Method	$\|(x_1 - x_0)\|$	$\|(x_2 - x_1)\|$	$\|(x_3 - x_2)\|$	$\|\zeta_7(x_3)\|$	*COC*
MSSV8	0.01000	4.6629×10^{-15}	9.0468×10^{-114}	5.4488×10^{-903}	8.0000
ACD8	0.01000	1.6809×10^{-15}	7.8996×10^{-118}	5.6400×10^{-936}	8.0000
LE8	0.01000	1.5489×10^{-17}	5.1353×10^{-138}	2.2496×10^{-1101}	8.0000
SH8	0.01000	5.5733×10^{-17}	5.0151×10^{-131}	6.4674×10^{-1043}	8.0000
BAC8	0.01000	1.0436×10^{-14}	1.6235×10^{-110}	1.6709×10^{-876}	8.0000
TKM9	0.01000	7.7009×10^{-16}	6.9025×10^{-129}	5.6202×10^{-1093}	8.5324
NWM9	0.01000	4.6629×10^{-15}	3.3257×10^{-128}	2.3441×10^{-1133}	8.8878

Table 8. Comparisons of without-memory and with-memory methods after first three (n = 3) iterations for $\zeta_8(x)$.

Method	$\|(x_1 - x_0)\|$	$\|(x_2 - x_1)\|$	$\|(x_3 - x_2)\|$	$\|\zeta_8(x_3)\|$	*COC*
MSSV8	0.01410	6.4479×10^{-16}	1.5052×10^{-122}	5.9152×10^{-974}	8.0000
ACD8	0.01410	3.2077×10^{-17}	7.2736×10^{-134}	2.2649×10^{-1065}	8.0000
LE8	0.01410	4.6217×10^{-17}	6.0900×10^{-133}	2.4665×10^{-1058}	8.0000
SH8	0.01410	1.2569×10^{-18}	5.3125×10^{-147}	2.4104×10^{-1172}	8.0000
BAC8	0.01410	2.1035×10^{-10}	2.6249×10^{-72}	6.8772×10^{-566}	8.0000
TKM9	0.01410	2.6040×10^{-16}	7.0174×10^{-135}	6.8920×10^{-1145}	8.5322
NWM9	0.01410	6.4479×10^{-16}	1.0519×10^{-137}	3.8029×10^{-1232}	9.0000

The numerical results in Tables 1–8 and Figure 1 show that the newly proposed with-memory method NWM9 is highly competitive and possesses fast convergence towards the roots with minimal absolute residual error and a minimum error value in consecutive iteration as compared to the other existing methods. Furthermore, the numerical findings show that the computational order of convergence aligns with the theoretical convergence order of the newly proposed method in the test functions.



(**a**) $\zeta_1(x)$ (**b**) $\zeta_2(x)$

(**c**) $\zeta_3(x)$ (**d**) $\zeta_4(x)$

(**e**) $\zeta_5(x)$ (**f**) $\zeta_6(x)$

(**g**) $\zeta_7(x)$ (**h**) $\zeta_8(x)$

Figure 1. Comparison of the methods based on the error in consecutive iterations, $|x_n - x_{n-1}|$, after the first three iterations.

4. Conclusions

In this article, we present a three-point with-memory iterative method using a self-accelerating parameter, elevating its convergence to the ninth order for nonlinear equations. We improve the efficiency index of an existing eighth-order method from EI = 1.6818 to EI = 1.7272 and raise its R-order of convergence from 8 to 8.8989 by adding this parameter, which is calculated using the Hermite interpolating polynomial, without the need for extra function evaluation. Though it has a greater convergence order than other known approaches, this approach not only speeds up convergence but also needs fewer function evaluations. Our results show that the recently proposed approach NWM9 is a very efficient option for solving nonlinear equations, providing better performance with faster convergence and smaller asymptotic constants.

Author Contributions: Conceptualisation, S.K.M. and S.P.; methodology, S.K.M. and S.P.; software, S.K.M., S.P., and L.J.; validation, S.K.M. and S.P.; formal analysis, S.P., S.K.M., and L.J.; resources, S.K.M.; writing—original draft preparation, S.P. and S.K.M.; writing—review and editing, S.K.M., S.P., and L.J.; visualization, S.P. and S.K.M.; supervision, S.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Pho, K.H. Improvements of the Newton–Raphson method. *J. Comput. Appl. Math.* **2022**, *408*, 114106. [CrossRef]
2. Traub, J.F. *Iterative Methods for the Solution of Equations*; Mathematical Association of America: Washington, DC, USA, 1982; Volume 312.
3. Liu, C.S.; Chang, C.W.; Kuo, C.L. Memory-Accelerating Methods for One-Step Iterative Schemes with Lie Symmetry Method Solving Nonlinear Boundary-Value Problem. *Symmetry* **2024**, *16*, 120. [CrossRef]
4. Sharma, E.; Mittal, S.K.; Jaiswal, J.P.; Panday, S. An Efficient Bi-Parametric with-Memory Iterative Method for Solving Nonlinear Equations. *AppliedMath* **2023**, *3*, 1019–1033. [CrossRef]
5. Thangkhenpau, G.; Panday, S.; Mittal, S.K. New Derivative-Free Families of Four-Parametric with and Without Memory Iterative Methods for Nonlinear Equations. In Proceedings of the International Conference on Science, Technology and Engineering, Coimbatore, India, 17–18 November 2023; Springer Nature: Singapore, 2023; pp. 313–324. [CrossRef]
6. Choubey, N.; Jaiswal, J.P.; Choubey, A. Family of multipoint with memory iterative schemes for solving nonlinear equations. *Int. J. Appl. Comput. Math.* **2022**, *8*, 83. [CrossRef]
7. Choubey, N.; Jaiswal, J.P. Two-and three-point with memory methods for solving nonlinear equations. *Numer. Anal. Appl.* **2017**, *10*, 74–89. [CrossRef]
8. Choubey, N.; Cordero, A.; Jaiswal, J.P.; Torregrosa, J.R. Dynamical techniques for analyzing iterative schemes with memory. *Complexity* **2018**, *2018*, 1232341. [CrossRef]
9. Erfanifar, R. A class of efficient derivative free iterative method with and without memory for solving nonlinear equations. *Comput. Math. Comput. Model. Appl.* **2022**, *1*, 20–26. [CrossRef]
10. Howk, C.L.; Hueso, J.L.; Martínez, E.; Teruel, C. A class of efficient high-order iterative methods with memory for nonlinear equations and their dynamics. *Math. Methods Appl. Sci.* **2018**, *41*, 7263–7282. [CrossRef]
11. Sharma, H.; Kansal, M.; Behl, R. An Efficient Two-Step Iterative Family Adaptive with Memory for Solving Nonlinear Equations and Their Applications. *Math. Comput. Appl.* **2022**, *27*, 97. [CrossRef]
12. Sharma, E.; Panday, S.; Mittal, S.K.; Joița, D.M.; Pruteanu, L.L.; Jäntschi, L. Derivative-free families of with-and without-memory iterative methods for solving nonlinear equations and their engineering applications. *Mathematics* **2023**, *11*, 4512. [CrossRef]
13. Wang, X.; Zhang, T. Some Newton-type iterative methods with and without memory for solving nonlinear equations. *Int. J. Comput. Methods* **2014**, *11*, 1350078. [CrossRef]
14. Liu, C.S.; Chang, C.W. New Memory-Updating Methods in Two-Step Newton's Variants for Solving Nonlinear Equations with High Efficiency Index. *Mathematics* **2024**, *12*, 581. [CrossRef]
15. Panday, S.; Mittal, S.K.; Stoenoiu, C.E.; Jäntschi, L. A New Adaptive Eleventh-Order Memory Algorithm for Solving Nonlinear Equations. *Mathematics* **2024**, *12*, 1809. [CrossRef]
16. Matthies, G.; Salimi, M.; Sharifi, S.; Varona, J.L. An optimal three-point eighth-order iterative method without memory for solving nonlinear equations with its dynamics. *Jpn. J. Ind. Appl. Math.* **2016**, *33*, 751–766. [CrossRef]

17. Ortega, J.M.; Rheinboldt, W.C. *Iterative Solution of Nonlinear Equations in Several Variables*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2000.
18. Alefeld, G.; Herzberger, J. *Introduction to Interval Computation*; Academic Press: Cambridge, MA, USA, 2012.
19. Abdullah, S.; Choubey, N.; Dara, S. Dynamical analysis of optimal iterative methods for solving nonlinear equations with applications. *J. Appl. Anal. Comput.* **2024**, *14*, 3349–3376.
20. Naseem, A.; Rehman, M.A.; Qureshi, S.; Ide, N.A.D. Graphical and numerical study of a newly developed root-finding algorithm and its engineering applications. *IEEE Access* **2023**, *11*, 2375–2383. [CrossRef]
21. Abdullah, S.; Choubey, N.; Dara, S. Optimal fourth-and eighth-order iterative methods for solving nonlinear equations with basins of attraction. *J. Appl. Math. Comput.* **2024**, *70*, 3477–3507. [CrossRef]
22. Lotfi, T.; Eftekhari, T. A New Optimal Eighth-Order Ostrowski-Type Family of Iterative Methods for Solving Nonlinear Equations. *Chin. J. Math.* **2014**, *2014*, 369713. [CrossRef]
23. Solaiman, O.S.; Hashim, I. Optimal Eighth-Order Solver for Nonlinear Equations with Applications in Chemical Engineering. *Intell. Autom. Soft Comput.* **2021**, *27*, 379–390. [CrossRef]
24. Behl, R.; Alshomrani, A.S.; Chun, C. A general class of optimal eighth-order derivative free methods for nonlinear equations. *J. Math. Chem.* **2020**, *58*, 854–867. [CrossRef]
25. Torkashv, V.; Kazemi, M.; Moccari, M. Structure a family of three-step with-memory methods for solving nonlinear equations and their dynamics. *Math. Anal. Convex Optim.* **2021**, *2*, 119–137.
26. Weerakoon, S.; Fernando, T.G.I. A variant of Newton's method with accelerated third-order convergence. *Appl. Math. Lett.* **2000**, *13*, 87–93. [CrossRef]