

Article

Extension of King's Iterative Scheme by Means of Memory for Nonlinear Equations

Saima Akram ^{1,2}, Maira Khalid ² , Moin-ud-Din Junjua ³ , Shazia Altaf ⁴ and Sunil Kumar ^{5,*} 

¹ Department of Mathematics, Government College Women University Faisalabad, Faisalabad 38000, Pakistan; saimaakram@gcwuf.edu.pk

² Centre for Advanced Studies in Pure and Applied Mathematics, Bahauddin Zakariya, Multan 60000, Pakistan; mairakhalid569@gmail.com

³ Department of Mathematics, Ghazi University, Dera Ghazi Khan 32200, Pakistan; moin_junjua@yahoo.com or mudin@gudgk.edu.pk

⁴ Department of Mathematics and Statistics, Institute of Southern Punjab, Multan 60800, Pakistan; shaz.altaf.sa@gmail.com

⁵ Department of Mathematics, University Centre for Research and Development, Chandigarh University, Mohali 140413, India

* Correspondence: sfageria1988@gmail.com

Abstract: We developed a new family of optimal eighth-order derivative-free iterative methods for finding simple roots of nonlinear equations based on King's scheme and Lagrange interpolation. By incorporating four self-accelerating parameters and a weight function in a single variable, we extend the proposed family to an efficient iterative scheme with memory. Without performing additional functional evaluations, the order of convergence is boosted from 8 to 15.51560, and the efficiency index is raised from 1.6817 to 1.9847. To compare the performance of the proposed and existing schemes, some real-world problems are selected, such as the eigenvalue problem, continuous stirred-tank reactor problem, and energy distribution for Planck's radiation. The stability and regions of convergence of the proposed iterative schemes are investigated through graphical tools, such as 2D symmetric basins of attractions for the case of memory-based schemes and 3D stereographic projections in the case of schemes without memory. The stability analysis demonstrates that our newly developed schemes have wider symmetric regions of convergence than the existing schemes in their respective domains.

Keywords: nonlinear equation; multipoint iterative methods; convergence order; with-memory method; efficiency index; polynomiography; stereographic projection



Citation: Akram, S.; Khalid, M.; Junjua, M.-u.D.; Altaf, S.; Kumar, S. Extension of King's Iterative Scheme by Means of Memory for Nonlinear Equations. *Symmetry* **2023**, *15*, 1116. <https://doi.org/10.3390/sym15051116>

Academic Editors: Sergei D. Odintsov and Calogero Vetro

Received: 27 April 2023

Revised: 13 May 2023

Accepted: 16 May 2023

Published: 19 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Solving nonlinear equations or root finding is an essential task in numerical analysis and has a wide range of applications in physics, chemistry, mathematical biology, medicines, economics, and engineering. Numerical or iterative methods to find approximate solutions of nonlinear equations are the most frequently used techniques. One can distinguish between two common approaches for numerical solutions of nonlinear equations, namely one-point and multipoint iterative schemes. Newton's method is one of the famous root-finding methods for solving a single nonlinear equation $f(x) = 0$, given as follows [1]:

$$x_m = x_m - \frac{f(x_m)}{f'(x_m)}, m \geq 0, \quad (1)$$

where f is a real valued function. Newton's scheme is a one-point method and is quadratically convergent in some neighborhood of the root η of f for an initial guess x_0 close enough to η . There are several two-point methods to find simple roots of $f(x) = 0$ in the literature.

Among them, King's method is the most popular two-point fourth-order method [2], which is given as follows:

$$\begin{aligned}y_m &= x_m - \frac{f(x_m)}{f'(x_m)}, m \geq 0, \\z_m &= y_m - \frac{f(x_m) + \gamma f(y_m)}{f(x_m) + (\gamma - 2)f(y_m)} \frac{f(y_m)}{f'(x_m)}, \gamma \in \mathbb{R}.\end{aligned}\quad (2)$$

Multipoint iterative schemes gained much interest at the beginning of the twenty-first century, since they overcome the theoretical limits of one-point methods regarding convergence speed and computational efficiency. Some root-finding schemes based on multipoint iterations are first studied in the books by Traub [1] and Ostrowski [3] and the papers in [2,4], published in the twentieth century. Multipoint schemes can be further classified into two categories: "methods with memory" and "methods without memory". A multipoint method using current information only is called a method without memory, while a method employing the current as well as previous information is known as a method with memory. An important aspect of the optimality of the methods with memory is that they improve the convergence order and efficiency index of the iterative schemes without memory without using additional function evaluations. Moreover, this approach makes it an impactful class of multipoint iterative methods. Kung and Traub conjectured that an iterative scheme without memory based on r evaluations of function could obtain an order of convergence of at most 2^{r-1} (known as optimal order). The computational efficiency index of an iterative method is expressed as $2^{\frac{r-1}{r}}$ [3].

Traub [1] proposed a uniparametric derivative-free scheme with memory by a slight modification of Steffensen's iterative method [5] for suitably chosen x_0, φ_0 as follows:

$$\begin{aligned}v_m &= x_m + \varphi_m f(x_m), m \geq 0, \\x_m &= x_m - \frac{f(x_m)}{f[v_m, x_m]}, \varphi_m = -\frac{1}{N_1'(x_m)},\end{aligned}\quad (3)$$

where $f[v, x] = \frac{f(v)-f(x)}{(v-x)}$ denotes divided difference of first order, $N_1 = f(x_m) + (x - x_m) f[v_m, x_m]$, and φ_m is the free accelerating parameter. The iterative scheme with memory (3) has a convergence order of 2.41. Several researchers have developed root-finding methods using memory based on existing optimal methods without memory; see, e.g., [2,6–16].

It is a common fact that iterative schemes are very sensitive towards an initial guess. To resolve this difficulty, we identify the region in which a root lies and, as a result, a variety of safe initial guesses are available to choose from. The convergence regions and stability of an iterative scheme can be visualized in 2D dynamical planes with the help of symmetric basins of attractions and drawing the 3D stereographic projections on the sphere.

One of the biggest challenges of iterative processes is to verify that they converge to the exact solution, rather than getting stuck in a local minimum or diverging. An effective way for examining the behavior of iterative methods, particularly those based on fixed-point iterations, is the analysis of basins of attractions. The areas of the parameter space that correspond to the same solution are referred to as basins of attractions. The conditions under which the iterative approach converges, as well as the rate of convergence and the stability of the solution, can be found by examining the basins of attractions. This information is essential for choosing the best iterative approach for a particular problem, as well as for adjusting a method's parameters to determine how well it performs.

Motivated and inspired by the research being conducted in this direction, we developed a new modified King's type scheme with memory, having a convergence order of at least 15.5156 based on Lagrange interpolation. In Section 2, we present a new optimal eighth-order derivative-free iterative family without memory. In Section 3, we extend the proposed optimal scheme without memory presented in Section 2 to an iterative scheme with memory and provide its convergence analysis. In Section 4, we present some particular

cases of the weight functions and iterative methods corresponding to the weight functions. In Section 5, we give applications of the presented iterative methods to solve engineering problems, such as the energy distribution for Planck's radiation and continuous stirred-tank reactor. A comparison of the newly developed iterative scheme with already existing similar schemes is also presented in Section 5 by using different test functions. In Section 6, we present the extensive stability analysis of the proposed scheme without memory by drawing its 3D stereographic projections. In Section 7, we present dynamical analysis of the proposed iterative scheme with memory with the help of its 2D symmetric basins of attractions. Finally, the conclusions are discussed in Section 8.

2. Extension of King's Method to an Optimal Eighth-Order Derivative-Free Scheme

Our primary aim is to develop an optimal eighth-order method without memory based on Lagrange interpolation which can be extended to an iterative scheme with memory. By adding Newton's method to the third step of King's method (2), we obtain

$$\begin{aligned}q_m &= x_m - \frac{f(x_m)}{f'(x_m)}, \\h_m &= q_m - \frac{f(q_m)}{f'(x_m)} \frac{f(x_m) + \omega f(q_m)}{f(x_m) + (\omega - 2)f(q_m)}, \\x_{m+1} &= h_m - \frac{f(h_m)}{f'(h_m)}.\end{aligned}\quad (4)$$

The three-step iterative scheme (4) is not optimal because it needs five evaluations of functions to provide eighth-order convergence. To make it optimal, derivative-free, and extendable to a with-memory scheme, we adopt the following procedure.

We use the approximation of the derivative $f'(x_m)$ in the first step of the iterative method (4) as follows:

$$f'(x_m) \approx f[v_m, x_m] + \beta_2 f(v_m), \quad (5)$$

where $v_m = x_m + \beta_1 f(x_m)$.

The derivative $f'(x_m)$ in the second step is replaced by the following approximation (with a weight function U , depending on variable $s_m = \frac{f(q_m)}{f(x_m)}$):

$$f'(x_m) \approx \frac{f[q_m, v_m] + \beta_2 f(v_m) + \beta_3 (q_m - v_m)(q_m - x_m)}{U(s_m)}. \quad (6)$$

We replace the derivative $f'(h_m)$ in the third step of the iterative scheme (4) by the following approximation:

$$f'(h_m) \approx L'_3(h_m) + \beta_4 (h_m - v_m)(h_m - q_m)(h_m - x_m), \quad (7)$$

where $\beta_1, \beta_2, \beta_3$, and β_4 are free parameters and

$$\begin{aligned}L'_3(h_m) &= f[h_m, v_m] + f[h_m, x_m] + f[h_m, q_m] - f[x_m, q_m] - f[v_m, q_m] - f[x_m, v_m] \\&+ \frac{(x_m - h_m)f(x_m)}{(x_m - v_m)(x_m - q_m)} + \frac{(v_m - h_m)f(v_m)}{(v_m - x_m)(v_m - q_m)} + \frac{(q_m - h_m)f(q_m)}{(q_m - v_m)(q_m - x_m)},\end{aligned}$$

where $L'_3(h_m)$ is the derivative of Lagrange interpolating polynomial of degree three that interpolates x_m, v_m, q_m , and h_m .

By using the above approximations, we obtain the following eighth-order iterative method without memory using four function evaluations per cycle:

$$\begin{aligned}
 v_m &= x_m + \beta_1 f(x_m), \\
 q_m &= x_m - \frac{f(x_m)}{f[v_m, x_m] + \beta_2 f(v_m)}, \\
 h_m &= q_m - U(s_m) \left(\frac{f(q_m)}{f[q_m, v_m] + \beta_2 f(v_m) + \beta_3 (q_m - v_m)(q_m - x_m)} \right) \\
 &\quad \left(\frac{f(x_m) + \omega f(q_m)}{f(x_m) + (\omega - 2)f(q_m)} \right), \\
 x_{m+1} &= h_m - \frac{f(h_m)}{R_m},
 \end{aligned} \tag{8}$$

where

$$\begin{aligned}
 R_m &= f[h_m, v_m] + f[h_m, x_m] + f[h_m, q_m] - f[x_m, q_m] - f[v_m, q_m] - f[x_m, v_m] \\
 &\quad + \frac{(x_m - h_m)f(x_m)}{(x_m - v_m)(x_m - q_m)} + \frac{(v_m - h_m)f(v_m)}{(v_m - x_m)(v_m - q_m)} + \frac{(q_m - h_m)f(q_m)}{(q_m - v_m)(q_m - x_m)} \\
 &\quad + \beta_4 (h_m - v_m)(h_m - q_m)(h_m - x_m).
 \end{aligned}$$

The next theorem indicates that the proposed method without memory (8) has a convergence order of eight with an efficiency index of $8^{\frac{1}{4}} \approx 1.68179$.

Theorem 1. Suppose that $f : I_f \subset \mathbb{R} \rightarrow \mathbb{R}$ is sufficiently differentiable and $\eta \in I_f$ is a real root of an equation $f(x) = 0$, where $I_f \subset \mathbb{R}$ is an open interval and x_0 is a good initial guess close to root η . Then, the convergence order of four parametric three-step schemes (8) is at least eight if the following conditions hold for the weight function $U(s_m)$:

$$U(0) = 1, U'(0) = -1, U''(0) = 2, \tag{9}$$

and the error equation of the iterative scheme (8) for all values of $\beta_1, \beta_2, \beta_3$, and β_4 is given as

$$\begin{aligned}
 e_{m+1} &= \frac{1}{f'(\eta)^2} ((C_2 + \beta_2)^2 (\beta_1 f'(\eta) + 1)^4 (2f'(\eta)^2 \omega c_2^2 \beta_1 + 4f'(\eta)^2 C_2 \beta_1 \beta_2 \omega \\
 &\quad + 2f'(\eta)^2 \beta_1 \beta_2^2 \omega - 2f'(\eta)^2 C_2^2 \beta_1 - 4\beta_2 f'(\eta)^2 \beta_1 C_2 - 2f'(\eta)^2 \beta_1 \beta_2^2 + 2f'(\eta) C_2^2 \omega \\
 &\quad + 4f'(\eta) C_2 \beta_2 \omega + 2f'(\eta)^2 \beta_2^2 \omega - 2\beta_2 f'(\eta) C_2 - 2f'(\eta) \beta_2^2 - f'(\eta) C_3 + \beta_3) \\
 &\quad (2f'(\eta)^2 \omega c_2^3 \beta_1 + 4f'(\eta)^2 \omega C_2^2 \beta_1 \beta_2 + 2f'(\eta)^2 \omega C_2 \beta_1 \beta_2^2 - 2f'(\eta)^2 C_2^3 \beta_1 \\
 &\quad - 4f'(\eta)^2 \beta_1 \beta_2 C_2^2 - 2f'(\eta)^2 C_2 \beta_1 \beta_2^2 + 2f'(\eta) \omega C_2^3 + 4f'(\eta) \omega C_2^2 \beta_2 + 2f'(\eta) \omega C_2 \beta_2^2 \\
 &\quad - 2f'(\eta) C_2^2 \beta_2 - 2f'(\eta) C_2 \beta_2^2 - f'(\eta) C_2 C_3 + f'(\eta) C_4 + C_2 \beta_3 \\
 &\quad - \beta_4)) e_m^8 + O(e_m^9).
 \end{aligned} \tag{10}$$

Proof. Let e_m be the error in the root η at m th step, specified as

$$e_m = x_m - \eta.$$

Expanding $f(x_m)$ using Taylor series, we obtain

$$f(x_m) = f'(\eta)(e_m + C_2 e_m^2 + C_3 e_m^3 + C_4 e_m^4 + C_5 e_m^5 + C_6 e_m^6 + C_7 e_m^7 + C_8 e_m^8) + O(e_m^9),$$

where

$$C_n = \frac{f^{(n)}(\eta)}{n! f'(\eta)}, n \geq \eta.$$

We obtain the error term $e_{m, \xi} = v_m - \eta$ by using Taylor’s expansion:

$$e_{m, \xi} = (1 + \beta_1 f'(\eta))e_m + \beta_1 f'(\eta)(C_2 e_m^2 + C_3 e_m^3 + C_4 e_m^4 + C_5 e_m^5 + C_6 e_m^6 + C_7 e_m^7 + C_8 e_m^8) + O(e_m^9).$$

Similarly, by applying Taylor’s series, we obtain $f(\eta_m)$, which is given as follows:

$$f(v_m) = f'(\eta)((1 + \beta_1 f'(\eta))e_m + \beta_1 f'(\eta)C_2 e_m^2 + \dots + O(e_m^9)).$$

Thus, the error term of q_m is

$$e_{m, w} = (C_2 + \beta_2)(1 + \beta_1 f'(\eta))e_m^2 + \dots + O(e_m^9),$$

where

$$e_{m, w} = q_m - \eta.$$

By applying Taylor’s expansion, we obtain $f(q_m)$, which is given as

$$f(q_m) = f'(\eta)[(C_2 + \beta_2)(1 + \beta_1 f'(\eta))e_m^2 + \dots + O(e_m^9)].$$

Again, by Taylor’s series, we obtained the error expression of h_m as follows:

$$\begin{aligned} e_{m, z} = & -(C_2 + \beta_2)(U(0) - 1)(f'(\eta)\beta_1 + 1)e_n^2 + (3\beta_1 f'(\eta)C_3 - U'(0)f'(\eta)^2 C_2^2 \beta_1 \\ & - U'(0)f'(\eta)^2 \beta_1^2 \beta_2^2 - 2U'(0)f'(\eta)C_2^2 \beta_1 - 2U'(0)f'(\eta)\beta_1 \beta_2^2 - U(0)f'(\eta)^2 C_2 \beta_1^2 \beta_2 \\ & - 2U(0)f'(\eta)C_2 \beta_1 \beta_2 - 2f'(\eta)\beta_1 \beta_2^2 f'(\eta)^2 \beta_1^2 \beta_2^2 - U'(0)C_2^2 - U'(0)\beta_2^2 \\ & - 2U'(0)f'(\eta)^2 C_2 \beta_1^2 \beta_2 - 4U'(0)f'(\eta)C_2 \beta_1 \beta_2 - 2\beta_2 C_2 - f'(\eta)^2 C_2 \beta_1^2 \beta_2 \\ & - 2\beta_2 f'(\eta)\beta_1 C_2 - 2C_2^2 + f'(\eta)^2 C_3 \beta_1^2 - 2\beta_1 f'(\eta)C_2^2 - \beta_1^2 f'(\eta)^2 C_2^2 + U(0)C_2^2 \\ & - 2U(0)C_3 - 2U'(0)C_2 \beta_2 - U(0)f'(\eta)^2 C_3 \beta_1^2 - 3U(0)f'(\eta)C_3 \beta_1 + 2C_3 \\ & - \beta_2^2 e_m^3 + \dots + O(e_m^9). \end{aligned} \tag{11}$$

In order to achieve fourth-order convergence, we choose $U(0) = 1, U'(0) = -1$, and $U''(0) = 2$; we have an error term of h_m :

$$\begin{aligned} e_{m, z} = & \frac{1}{f'(\eta)}((f'(\eta)\beta_1 + 1)^2(C_2 + \beta_2)(2f'(\eta)^2 \omega C_2^2 \beta_1 + 4f'(\eta)^2 C_2 \beta_1 \beta_2 \omega + 2f'(\eta)^2 \beta_1 \beta_2^2 \omega \\ & - 2f'(\eta)^2 C_2^2 \beta_1 - 4\beta_2 f'(\eta)^2 \beta_1 C_2 - 2f'(\eta)^2 \beta_1 \beta_2^2 + 2f'(\eta)C_2^2 \omega + 4f'(\eta)C_2 \beta_2 \omega \\ & + 2f'(\eta)^2 \beta_2^2 \omega - 2\beta_2 f'(\eta)C_2 - 2f'(\eta)\beta_2^2 - f'(\eta)C_3 + \beta_3)e_m^4 + \dots + O(e_m^9). \end{aligned}$$

By using the Taylor’s series, we can easily find the expression of $f(h_m)$ and R_m that is given in the newly proposed with-memory method (8). Thus, we obtain a convergence order of eight in the proposed iterative scheme (8) as follows:

$$\begin{aligned} e_{m+1} = & \frac{1}{f'(\eta)^2}((C_2 + \beta_2)^2(\beta_1 f'(\eta) + 1)^4(2f'(\eta)^2 \omega C_2^2 \beta_1 + 4f'(\eta)^2 C_2 \beta_1 \beta_2 \omega \\ & + 2f'(\eta)^2 \beta_1 \beta_2^2 \omega - 2f'(\eta)^2 C_2^2 \beta_1 - 4\beta_2 f'(\eta)^2 \beta_1 C_2 - 2f'(\eta)^2 \beta_1 \beta_2^2 + 2f'(\eta)C_2^2 \omega \\ & + 4f'(\eta)C_2 \beta_2 \omega + 2f'(\eta)^2 \beta_2^2 \omega - 2\beta_2 f'(\eta)C_2 - 2f'(\eta)\beta_2^2 - f'(\eta)C_3 + \beta_3) \\ & (2f'(\eta)^2 \omega C_2^3 \beta_1 + 4f'(\eta)^2 \omega C_2^2 \beta_1 \beta_2 + 2f'(\eta)^2 \omega C_2 \beta_1 \beta_2^2 - 2f'(\eta)^2 C_2^3 \beta_1 \\ & - 4f'(\eta)^2 \beta_1 \beta_2 C_2^2 - 2f'(\eta)^2 C_2 \beta_1 \beta_2^2 + 2f'(\eta)\omega C_2^3 + 4f'(\eta)\omega C_2^2 \beta_2 + 2f'(\eta)\omega C_2 \beta_2^2 \\ & - 2f'(\eta)C_2^2 \beta_2 - 2f'(\eta)C_2 \beta_2^2 - f'(\eta)C_2 C_3 + f'(\eta)C_4 + C_2 \beta_3 \\ & - \beta_4))e_m^8 + O(e_m^9). \end{aligned} \tag{12}$$

□

Remark 1. The above theorem verifies that the proposed method (8) has eighth-order convergence with an efficiency index of $8^{\frac{1}{4}} \approx 1.68179$. Furthermore, note that if we chose $\beta_1 = -\frac{1}{f'(\eta)}$ and $\beta_2 = -C_2$, then the coefficient of e_m^8 vanishes and we obtain the following error equation:

$$e_{m+1} = \left[C_2^4 C_3^3 (\beta_3 - f'(\eta)C_3) (C_2 (\beta_3 - f'(\eta)C_3) + f'(\eta)C_4 - \beta_4) \right] \frac{e_m^{14}}{f''(\eta)} + O(e_m^{15}).$$

By choosing $\beta_3 = f'(\eta)C_3$ and $\beta_4 = f'(\eta)C_4$, the resulting scheme has an optimal order of convergence of at least sixteen. In order to accomplish the with-memory iterative scheme, parameters play an essential role. Thus, from the above error analysis, we observe that the multipoint scheme (8) is extendable to an iterative scheme with memory.

3. Efficient King’s Type Scheme With Memory

This section covers a significant role in our work. Here, we extend the proposed optimal scheme (8) such that it achieves the highest possible order and efficiency by using previous iteration values. We use four self-accelerating parameters, $\beta_1, \beta_2, \beta_3$, and β_4 in the scheme (8) and approximate these parameters by Newton’s interpolatory polynomial of the appropriate degree. So, in this way, we can increase the order of convergence. If we choose $\beta_1 = -\frac{1}{f'(\eta)}, \beta_2 = -C_2, \beta_3 = f'(\eta)C_3$ and $\beta_4 = f'(\eta)C_4$, where $C_j = \frac{f^{(j)}(\eta)}{j!f'(\eta)}$ and $j \geq 2$, then the order of convergence reaches up to sixteen. Thus, to obtain an iterative scheme with memory, the parameters $\beta_1, \beta_2, \beta_3$, and β_4 are determined by using the following formulas:

$$\begin{aligned} \beta_1 = \beta_{1,m} &= -\frac{1}{N_4'(x_m)} \approx -\frac{1}{f'(\eta)}, \\ \beta_2 = \beta_{2,m} &= -\frac{N_5''(v_m)}{2N_5'(v_m)} \approx -C_2, \\ \beta_3 = \beta_{3,m} &= -\frac{N_6'''(q_m)}{6} \approx f'(\eta)C_3, \\ \beta_4 = \beta_{4,m} &= -\frac{N_7^{iv}(h_m)}{24} \approx f'(\eta)C_4. \end{aligned} \tag{13}$$

The above approximations are made by using Newton’s fourth-, fifth-, sixth-, and seventh-degree interpolatory polynomials, passing through the best available approximations, and they are given as follows:

$$\begin{aligned} N_4(\varphi) &= N_4(\varphi; x_m, h_m, q_m, v_m, x_m), \\ N_5(\varphi) &= N_5(\varphi; v_m, x_m, h_m, q_m, v_m, x_m), \\ N_6(\varphi) &= N_6(\varphi; q_m, v_m, x_m, h_m, q_m, v_m, x_m), \\ N_7(\varphi) &= N_7(\varphi; h_m, q_m, v_m, x_m, h_m, q_m, v_m, x_m), \end{aligned}$$

for any $m \geq 1$. The explicit representations for $N_4(\varphi), N_5(\varphi), N_6(\varphi)$, and $N_7(\varphi)$ are given as follows:

$$\begin{aligned} &N_4(\varphi; x_m, h_m, q_m, v_m, x_m) \\ &= f(x_m) + f[x_m, h_m](\varphi - x_m) + f[x_m, h_m, q_m](\varphi - x_m)(\varphi - h_m) \\ &\quad + f[x_m, h_m, q_m, v_m](\varphi - x_m)(\varphi - h_m)(\varphi - q_m) \\ &\quad + f[x_m, h_m, q_m, v_m, x_m](\varphi - x_m)(\varphi - h_m)(\varphi - q_m) \\ &\quad (\varphi - v_m), \end{aligned} \tag{14}$$

$$\begin{aligned}
 & N_5(\varphi; v_m, x_m, h_m, q_m, v_m, x_m) \\
 = & f(v_m) + f[v_m, x_m](\varphi - v_m) + f[v_m, x_m, h_m](\varphi - v_m)(\varphi - x_m) \\
 & + f[v_m, x_m, h_m, q_m](\varphi - v_m)(\varphi - x_m)(\varphi - h_m) + f[v_m, x_m, h_m, q_m, v_m] \\
 & (\varphi - v_m)(\varphi - x_m)(\varphi - h_m)(\varphi - q_m) + f[v_m, x_m, h_m, q_m, v_m, x_m] \\
 & (\varphi - v_m)(\varphi - x_m)(\varphi - h_m)(\varphi - q_m)(\varphi - v_m), \tag{15}
 \end{aligned}$$

$$\begin{aligned}
 & N_6(\varphi; q_m, v_m, x_m, h_m, q_m, v_m, x_m) \\
 = & f(q_m) + f[q_m, v_m](\varphi - q_m) + f[q_m, v_m, x_m](\varphi - q_m)(\varphi - v_m) \\
 & + f[q_m, v_m, x_m, h_m](\varphi - q_m)(\varphi - v_m)(\varphi - x_m) + f[q_m, v_m, x_m, h_m, q_m] \\
 & (\varphi - q_m)(\varphi - v_m)(\varphi - x_m)(\varphi - h_m) + f[q_m, v_m, x_m, h_m, q_m, v_m] \\
 & (\varphi - q_m)(\varphi - v_m)(\varphi - x_m)(\varphi - h_m)(\varphi - q_m) \\
 & + f[q_m, v_m, x_m, h_m, q_m, v_m, x_m](\varphi - q_m)(\varphi - v_m)(\varphi - x_m) \\
 & (\varphi - h_m)(\varphi - q_m)(\varphi - v_m), \tag{16}
 \end{aligned}$$

$$\begin{aligned}
 & N_7(\varphi; h_m, q_m, v_m, x_m, h_m, q_m, v_m, x_m) \\
 = & f(h_m) + f[h_m, q_m](\varphi - h_m) + f[h_m, q_m, \eta_x](\varphi - h_m)(\varphi - q_m) \\
 & + f[h_m, q_m, v_m, x_m](\varphi - h_m)(\varphi - q_m)(\varphi - \eta_x) + f[h_m, q_m, v_m, x_m, h_m] \\
 & (\varphi - h_m)(\varphi - q_m)(\varphi - v_m)(\varphi - x_m) + f[h_m, q_m, v_m, x_m, h_m, q_m] \\
 & (\varphi - h_m)(\varphi - q_m)(\varphi - v_m)(\varphi - x_m)(\varphi - h_m) \\
 & + f[h_m, q_m, v_m, x_m, h_m, q_m, v_m](\varphi - h_m)(\varphi - q_m)(\varphi - v_m)(\varphi - x_m) \\
 & (\varphi - h_m)(\varphi - q_m) + f[h_m, q_m, v_m, x_m, h_m, q_m, v_m, x_m] \\
 & (\varphi - h_m)(\varphi - q_m)(\varphi - v_m)(\varphi - x_m)(\varphi - h_m)(\varphi - q_m) \\
 & (\varphi - v_m). \tag{17}
 \end{aligned}$$

Now, we substitute the free parameters $\beta_1, \beta_2, \beta_3,$ and β_4 in (8) with $\beta_{1,x}, \beta_{2,m}, \beta_{3,m},$ and $\beta_{4,m}$. Consequently, we obtain a three-step iterative scheme with memory, as given below:

$$\begin{aligned}
 v_m &= x_m + \beta_{1,m}f(x_m), \beta_{1,m} = -\frac{1}{N'_4(x_m)}, m \geq 0, \\
 q_m &= x_m - \frac{f(x_m)}{f[v_m, x_m] + \beta_{2,m}f(v_m)}, \beta_{2,m} = -\frac{N''_5(v_m)}{2N'_5(v_m)}, \\
 \beta_{3,m} &= -\frac{N'''_6(q_m)}{6}, \beta_{4,m} = -\frac{N^{iv}_7(h_m)}{24}, \\
 h_m &= y_m - U(s_m) \left(\frac{f(q_m)}{f[q_m, v_m] + \beta_{2,m}f(v_m) + \beta_{3,m}(q_m - v_m)(q_m - x_m)} \right) \\
 & \quad \left(\frac{f(x_m) + \omega f(q_m)}{f(x_m) + (\omega - 2)f(q_m)} \right), \\
 x_m &= h_m - \frac{f(h_m)}{R_m}, \tag{18}
 \end{aligned}$$

where $\beta_{1,0}, \beta_{2,0}, \beta_{3,0},$ and $\beta_{4,0}$ should be chosen suitably, $s_m = \frac{f(q_m)}{f(x_m)}$, and $R_m = f[h_m, v_m] + f[h_m, x_m] + f[h_m, q_m] - f[x_m, q_m] - f[v_m, q_m] - f[x_m, v_m] + \frac{(x_m-h_m)f(x_m)}{(x_m-v_m)(x_m-q_m)} + \frac{(v_m-h_m)f(v_m)}{(v_m-x_m)(v_m-q_m)} + \frac{(q_m-h_m)f(q_m)}{(q_m-v_m)(q_m-x_m)} + \beta_{4,m}(h_m - v_m)(h_m - q_m)(h_m - x_m)$.

Lemma 1. If $\beta_{1,m} = -\frac{1}{N_4'(x_m)}, \beta_{2,m} = -\frac{N_5''(v_m)}{2N_5'(v_m)}, \beta_{3,m} = -\frac{N_6'''(q_m)}{6}, \beta_{4,m} = -\frac{N_7^{iv}(h_m)}{24}$, for $m = 1, 2, \dots$, then the following holds:

$$1 + \beta_{1,m}f'(\eta) \sim e_{m-1,z}e_{m-1,w}e_{m-1,\zeta}e_{m-1}, \tag{19}$$

$$c_2 + \beta_{2,m} \sim e_{m-1,z}e_{m-1,w}e_{m-1,\zeta}e_{m-1},$$

$$P_m \sim e_{m-1,z}e_{m-1,w}e_{m-1,\zeta}e_{m-1}, \tag{20}$$

$$Q_m \sim e_{m-1,z}e_{m-1,w}e_{m-1,\zeta}e_{m-1}, \tag{21}$$

where

$$P_m = (2f'(\eta)^2\omega c_2^2\beta_1 + 4f'(\eta)^2c_2\beta_1\beta_2\omega + 2f'(\eta)\beta_1^2\beta_2^2\omega - 2f'(\eta)^2C_2^2\beta_1 - 4\beta_2f'(\eta)^2\beta_1C_2 - 2f'(\eta)^2\beta_1\beta_2^2 + 2f'(\eta)C_2^2\omega + 4f'(\eta)C_2\beta_2\omega + 2f'(\eta)^2\beta_2^2\omega - 2\beta_2f'(\eta)C_2 - 2f'(\eta)\beta_2^2 - f'(\eta)C_3 + \beta_3), \tag{22}$$

$$Q_m = (2f'(\eta)^2\omega C_2^3\beta_1 + 4f'(\eta)^2\omega C_2^2\beta_1\beta_2 + 2f'(\eta)^2\omega C_2\beta_1\beta_2^2 - 2f'(\eta)^2C_2^3\beta_1 - 4\beta_2f'(\eta)^2\beta_1C_2^2 - 2f'(\eta)^2C_2\beta_1\beta_2^2 + 2f'(\eta)\omega C_2^3 + 4f'(\eta)\omega C_2^2\beta_2 + 2f'(\eta)\omega C_2\beta_2^2 - 2f'(\eta)C_2^2\beta_2 - 2f'(\eta)C_2\beta_2^2 - f'(\eta)C_2C_3 + f'(\eta)C_4 + C_2\beta_3 - r_4). \tag{23}$$

Proof. The proof follows from Lemma 1 in [15]. □

Theorem 2. Suppose that x_0 is an initial approximation which is close to a root η of a non-linear function $f(x)$, with f being sufficiently differentiable. If the self-accelerating parameters $\beta_{1,m}, \beta_{2,m}, \beta_{3,m}$, and $\beta_{4,m}$ are evaluated by the formulae given in (13), then the three-step with-memory scheme (18) has an R-order of convergence of at least 15.5156 with an efficiency index of $15.5156^{\frac{1}{3}} \approx 1.9847$.

Proof. Assume that the sequence $\{x_m\}$ is generated by the iterative scheme (18) converging to a real zero η with at least order r , then we may write

$$e_{m+1} \sim e_m^r, \tag{24}$$

where $e_m = x_m - \eta$. From (24), we obtain:

$$e_{m+1} \sim (e_m^r)^r = e_{m-1}^{r^2}. \tag{25}$$

Let $\{v_m\}, \{q_m\}$, and $\{h_m\}$ be the iterative sequences of the convergence orders σ_1, σ_2 , and σ_3 , then the error relation is of the form:

$$e_{m,\zeta} \sim (e_m^r)^{\sigma_1} = e_{m-1}^{r\sigma_1}, \tag{26}$$

$$e_{m,w} \sim (e_m^r)^{\sigma_2} = e_{m-1}^{r\sigma_2}, \tag{27}$$

$$e_{m,z} \sim (e_m^r)^{\sigma_3} = e_{m-1}^{r\sigma_3}. \tag{28}$$

Now, using (26)–(28) and Lemma 1, we obtain

$$1 + \beta_{1,m}f'(\eta) \sim e_{m-1}^{\sigma_1+\sigma_2+\sigma_3+1}, \quad (29)$$

$$C_2 + \beta_{2,m} \sim e_{m-1}^{\sigma_1+\sigma_2+\sigma_3+1}, \quad (30)$$

$$P_m \sim e_{m-1}^{\sigma_1+\sigma_2+\sigma_3+1}, \quad (31)$$

$$Q_m \sim e_{m-1}^{\sigma_1+\sigma_2+\sigma_3+1}, \quad (32)$$

where P_m and Q_m are given in (22) and (23). So, by Theorem 1, we obtain the following terms:

$$e_{m,\xi} \sim (1 + \beta_{1,m}f'(\eta))e_m, \quad (33)$$

$$e_{m,w} \sim (1 + \beta_{1,m}f'(\eta))(C_2 + \beta_{2,m})e_m^2, \quad (34)$$

$$e_{m,z} \sim \left[(1 + \beta_{1,m}f'(\eta))^2 (C_2 + \beta_{2,m}) P_m \right] e_m^4, \quad (35)$$

$$e_{m+1} \sim \left[(1 + \beta_{1,m}f'(\eta))^4 (C_2 + \beta_{2,m})^2 P_m Q_m \right] e_m^8, \quad (36)$$

where P_m and Q_m are given in (22) and (23). Now, substitute (24) and (29)–(32) in (33)–(36); we obtain

$$e_{m,\xi} = e_{m-1}^{\sigma_1+\sigma_2+\sigma_3+1+r}, \quad (37)$$

$$e_{m,w} = e_{m-1}^{2(\sigma_1+\sigma_2+\sigma_3+1)+2r}, \quad (38)$$

$$e_{m,z} = e_{m-1}^{4(\sigma_1+\sigma_2+\sigma_3+1)+4r}, \quad (39)$$

$$e_{m+1} = e_{m-1}^{8(\sigma_1+\sigma_2+\sigma_3+1)+8r}. \quad (40)$$

By attributing the coefficients of the suitable exponents of e_{m-1} in the pair of following relations (26)∧(37), (27)∧(38), (28)∧(39), and (25)∧(40), respectively, we obtained the system of equations in $\sigma_1, \sigma_2, \sigma_3$, and r as

$$\begin{aligned} r\sigma_1 - r - (\sigma_1 + \sigma_2 + \sigma_3 + 1) &= 0, \\ r\sigma_2 - 2r - 2(\sigma_1 + \sigma_2 + \sigma_3 + 1) &= 0, \\ r\sigma_3 - 4r - 4(\sigma_1 + \sigma_2 + \sigma_3 + 1) &= 0, \\ r^2 - 8r - 8(\sigma_1 + \sigma_2 + \sigma_3 + 1) &= 0. \end{aligned} \quad (41)$$

By solving these equations, we obtain the nontrivial solutions $\sigma_1 = 1.939451$, $\sigma_2 = 3.878902$, $\sigma_3 = 7.757804$, and $r = 15.515609$. Hence, the R-order of convergence of the proposed iterative scheme with memory is at least 15.5156. \square

4. Some Special Cases of Weight Functions

We choose the following choices of weight functions satisfying the conditions $U(0) = 1, U'(0) = -1, U''(0) = 2$:

$$U(s_m) = 1 - s_m + 2 \sin(s_m)^2,$$

$$U(s_m) = \frac{1}{1 + s_m - 2s_m^2}. \quad (42)$$

By using the above weight functions defined in (42), we obtain the following special cases of the proposed without-memory iterative scheme (8):

Method AK1:

$$\begin{aligned}
 v_m &= x_m + \beta_{1,m}f(x_m), \\
 q_m &= x_m - \frac{f(x_m)}{f[v_m, x_m] + \beta_{2,m}f(v_m)}, \\
 h_m &= q_m - (1 - s_m + 2 \sin(s_m)^2) \left(\frac{f(q_m)}{f[q_m, v_m] + \beta_{2,m}f(v_m) + \beta_{3,m}(q_m - v_m)(q_m - x_m)} \right) \\
 &\quad \left(\frac{f(x_m) + \omega f(q_m)}{f(x_m) + (\omega - 2)f(q_m)} \right), \\
 x_{m+1} &= h_m - \frac{f(h_m)}{R_m},
 \end{aligned} \tag{43}$$

Method AK2:

$$\begin{aligned}
 v_m &= x_m + \beta_{1,m}f(x_m), \\
 q_m &= x_m - \frac{f(x_m)}{f[v_m, x_m] + \beta_{2,m}f(v_m)}, \\
 h_m &= q_m - \left(\frac{1}{1 + s_m - 2s_m^2} \right) \left(\frac{f(q_m)}{f[q_m, v_m] + \beta_{2,m}f(v_m) + \beta_{3,m}(q_m - v_m)(q_m - x_m)} \right) \\
 &\quad \left(\frac{f(x_m) + \omega f(q_m)}{f(x_m) + (\omega - 2)f(q_m)} \right), \\
 x_{m+1} &= h_m - \frac{f(h_m)}{R_m}.
 \end{aligned} \tag{44}$$

The special cases of the proposed with-memory iterative scheme (18) are as follows:

Method AKM1:

$$\begin{aligned}
 v_m &= x_m + \beta_{1,m}f(x_m), \beta_{1,m} = -\frac{1}{N'_4(x_m)}, \\
 q_m &= x_m - \frac{f(x_m)}{f[v_m, x_m] + \beta_{2,m}f(v_m)}, \beta_{2,m} = -\frac{N''_5(v_m)}{2N'_5(v_m)}, \\
 \beta_{3,m} &= -\frac{N'''_6(q_m)}{6}, \beta_{4,m} = -\frac{N^{iv}_7(h_m)}{24}, \\
 h_m &= q_m - (1 - s_m + 2 \sin(s_m)^2) \left(\frac{f(q_m)}{f[q_m, v_m] + \beta_{2,m}f(v_m) + \beta_{3,m}(q_m - v_m)(q_m - x_m)} \right) \\
 &\quad \left(\frac{f(x_m) + \omega f(q_m)}{f(x_m) + (\omega - 2)f(q_m)} \right), \\
 x_{m+1} &= h_m - \frac{f(h_m)}{R_m}.
 \end{aligned} \tag{45}$$

Method AKM2:

$$\begin{aligned}
 v_m &= x_m + \beta_{1,m}f(x_m), \beta_{1,m} = -\frac{1}{N'_4(x_m)}, \\
 q_m &= x_m - \frac{f(x_m)}{f[v_m, x_m] + \beta_{2,m}f(v_m)}, \beta_{2,m} = -\frac{N''_5(v_m)}{2N'_5(v_m)}, \\
 \beta_{3,m} &= -\frac{N'''_6(q_m)}{6}, \beta_{4,m} = -\frac{N^{iv}_7(h_m)}{24}, \\
 h_m &= q_m - \left(\frac{1}{1 + s_m - 2s_m^2} \right) \left(\frac{f(q_m)}{f[q_m, v_m] + \beta_{2,m}f(v_m) + \beta_{3,m}(q_m - v_m)(q_m - x_m)} \right) \\
 &\quad \left(\frac{f(x_m) + \omega f(q_m)}{f(x_m) + (\omega - 2)f(q_m)} \right), \\
 x_{m+1} &= h_m - \frac{f(h_m)}{R_m},
 \end{aligned} \tag{46}$$

where $s_m = \frac{f(q_m)}{f(x_m)}$ and

$$R_m = f[h_m, v_m] + f[h_m, x_m] + f[h_m, q_m] - f[x_m, q_m] - f[v_m, q_m] - f[x_m, v_m] + \frac{(x_m - h_m)f(x_m)}{(x_m - v_m)(x_m - q_m)} + \frac{(v_m - h_m)f(v_m)}{(v_m - x_m)(v_m - q_m)} + \frac{(q_m - h_m)f(q_m)}{(q_m - v_m)(q_m - x_m)} + \beta_{4,m}(h_m - v_m)(h_m - q_m)(h_m - x_m).$$

5. Numerical Experiments

The primary purpose of the development of an iterative scheme for the solution of nonlinear equations is to achieve as rapid as possible convergence order with minimal cost of computation. In this section, we compare the newly constructed schemes with already existing methods. The computational order of convergence (COC) is defined by

$$COC \approx \frac{\log|f(x_{m+1})/f(x_m)|}{\log|f(x_m)/f(x_{m-1})|}.$$

The computer programming package, Maple 18, is used with 2000 fixed floating points in the numerical tests. The error approximation ($|x_m - \eta|$) for the first three iterations is given in Tables 1–10. We compare the family of without-memory methods (AK1) and (AK2) and with-memory methods (AKM1) and (AKM2) with the three-step optimal methods proposed by Lotfi et al. [17], represented by (LM); Cordero et al. [18], represented by (M1); and Zafar et al. [13], represented by (FWM), which are given below:

Method LM:

$$\begin{aligned} v_m &= x_m + \alpha_{1,m}f(x_m), \alpha_{1,m} = -\frac{1}{N'_4(x_m)}, m \geq 2, \\ q_m &= x_m - \frac{f(x_m)}{f[x_m, v_m] + \alpha_{2,m}f(v_m)}, \alpha_{2,m} = -\frac{N''_5(v_m)}{2N'_5(v_m)}, \\ \alpha_{3,m} &= \frac{N'''_6(q_m)}{6}, \alpha_{4,m} = \frac{N^{iv}_7(h_m)}{24}, \\ h_m &= q_m - \frac{f(q_m)}{f[q_m, x_m] + f[v_m, x_m, q_m](q_m - x_m) + \alpha_{3,m}(q_m - v_m)(q_m - x_m)}, \\ x_{m+1} &= h_m - \frac{f(h_m)}{B_m}, \end{aligned} \tag{47}$$

where $x_0, \alpha_{1,0}, \alpha_{2,0}, \alpha_{3,0}$, and $\alpha_{4,0}$ are given and

$$B_m = f[x_m, h_m] + (f[v_m, x_m, q_m] - f[v_m, x_m, h_m] - f[q_m, x_m, h_m])(x_m - h_m) + \alpha_{4,m}(h_m - v_m)(h_m - q_m)(h_m - x_m).$$

Method M1:

$$\begin{aligned} v_m &= x_m + p_{1,m}f(x_m), p_{1,m} = -\frac{1}{N'_4(x_m)}, m \geq 0, \\ q_m &= x_m - \frac{f(x_m)}{f[x_m, v_m] + p_{2,m}f(v_m)}, p_{2,m} = -\frac{N''_5(v_m)}{2N'_5(v_m)}, \\ p_{3,m} &= \frac{N'''_6(q_m)}{6}, p_{4,m} = \frac{N^{iv}_7(h_m)}{24}, d_m = \frac{f(q_m)}{f(x_m)}, \\ h_m &= q_m - (1 + 2d_m)(1 - d_m) \left(\frac{f(q_m)}{f[q_m, v_m] + p_{2,m}f(v_m) + p_{3,m}(q_m - v_m)(q_m - x_m)} \right), \\ x_{m+1} &= h_m - \frac{f(h_m)}{G_m}, \end{aligned} \tag{48}$$

where $p_{1,0}, p_{2,0}, p_{3,0}$, and $p_{4,0}$ are given and

$$G_m = f[q_m, h_m] + f[h_m, q_m, x_m](h_m - q_m) + f[h_m, q_m, x_m, v_m](h_m - q_m)(h_m - x_m) + p_{4,m}(h_m - v_m)(h_m - q_m)(h_m - x_m).$$

Method FWM:

$$\begin{aligned} v_m &= x_m + b_{1,x}f(x_m), b_{1,m} = -\frac{1}{N'_4(x_m)}, m \geq 0, \\ q_m &= x_m - \frac{f(x_m)}{f[x_m, v_m] + b_{2,m}f(v_m)}, b_{2,m} = -\frac{N''_5(v_m)}{2N'_5(v_m)}, \\ b_{3,m} &= \frac{N'''_6(q_m)}{6}, b_{4,m} = \frac{N^{iv}_7(h_m)}{24}, d_m = \frac{f(q_m)}{f(x_m)}, \\ h_m &= q_m - (1 - d_m) \left(\frac{f(q_m)}{f[q_m, v_m] + b_{2,m}f(v_m) + b_{3,m}(q_m - v_m)(q_m - x_m)} \right) \\ &\quad \left(\frac{f(x_m) + \omega f(q_m)}{f(x_m) + (\omega - 2)f(q_m)} \right), \\ x_{m+1} &= h_m - \frac{f(h_m)}{E_m}, \end{aligned} \tag{49}$$

where $b_{1,0}, b_{2,0}, b_{3,0}$, and $b_{4,0}$ are given and

$$E_m = f[q_m, h_m] + f[h_m, q_m, x_m](h_m - q_m) + f[h_m, y_x, x_m, v_m](h_m - q_m)(h_m - x_m) + b_{4,m}(h_m - v_m)(h_m - q_m)(h_m - x_m).$$

For the comparison of the with-memory schemes, we use $\beta_{1,0} = \beta_{3,0} = \beta_{4,0} = b_{1,0} = b_{3,0} = b_{4,0} = \alpha_{1,0} = \alpha_{4,0} = p_{1,0} = p_{3,0} = p_{4,0} = 0.01$, $\alpha_{2,0} = -0.1$, and $\beta_{2,0} = b_{2,0} = p_{2,0} = \alpha_{3,0} = 0.1$ to start the iterative process.

Table 1. Numerical results of methods without memory for $F_1(x)$.

| $F_1(x) = \exp(-x) - 1 + \frac{1}{5}x, x_0 = -0.16, \eta = 0$ | | | | |
|---|-----------------------|------------------------|-------------------------|------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 2.79×10^{-9} | 2.60×10^{-71} | 1.44×10^{-567} | 7.99 |
| M1 | 5.84×10^{-8} | 3.78×10^{-59} | 1.16×10^{-468} | 7.99 |
| FWM | 5.84×10^{-8} | 3.78×10^{-59} | 1.16×10^{-468} | 7.99 |
| AK1 | 1.95×10^{-8} | 1.89×10^{-63} | 1.47×10^{-503} | 7.99 |
| AK2 | 1.08×10^{-8} | 6.47×10^{-66} | 1.11×10^{-523} | 7.99 |

Table 2. Numerical results of methods with memory for $F_1(x)$.

| $F_1(x) = \exp(-x) - 1 + \frac{1}{5}x, x_0 = -0.16, \eta = 0$ | | | | |
|---|-----------------------|-------------------------|--------------------------|-------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 2.79×10^{-9} | 3.09×10^{-142} | 3.45×10^{-2000} | 13.97 |
| M1 | 5.84×10^{-8} | 3.44×10^{-120} | 1.35×10^{-1866} | 15.56 |
| FWM | 5.84×10^{-8} | 3.44×10^{-120} | 1.35×10^{-1866} | 15.56 |
| AKM1 | 1.95×10^{-8} | 7.52×10^{-128} | 2.29×10^{-1985} | 15.56 |
| AKM2 | 1.08×10^{-8} | 5.43×10^{-132} | 1.80×10^{-2000} | 15.15 |

Table 3. Numerical results of methods without memory for $F_2(x)$.

| $F_2(x) = x^9 - 29x^8 + 349x^7 - 2261x^6 + 8455x^5 - 17663x^4$ $+15927x^3 + 6993x^2 - 24732x + 12960, x_0 = 1.09, \eta = 1$ | | | | |
|--|-----------------------|------------------------|-------------------------|------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 1.85×10^{-4} | 9.40×10^{-23} | 4.05×10^{-169} | 8.00 |
| M1 | 3.24×10^{-3} | 1.72×10^{-8} | 4.04×10^{-51} | 8.08 |
| FWM | 3.24×10^{-3} | 1.72×10^{-8} | 4.04×10^{-51} | 8.08 |
| AK1 | 1.43×10^{-5} | 9.45×10^{-29} | 3.43×10^{-214} | 8.00 |
| AK2 | 1.55×10^{-4} | 9.50×10^{-23} | 1.82×10^{-168} | 8.00 |

Table 4. Numerical results of methods with memory for $F_2(x)$.

| $F_2(x) = x^9 - 29x^8 + 349x^7 - 2261x^6 + 8455x^5 - 17663x^4$ $+15927x^3 + 6993x^2 - 24732x + 12960, x_0 = 1.09, \eta = 1$ | | | | |
|--|-----------------------|------------------------|--------------------------|-------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 1.85×10^{-4} | 5.87×10^{-57} | 4.16×10^{-881} | 15.69 |
| M1 | 3.24×10^{-3} | 1.14×10^{-36} | 1.59×10^{-569} | 15.92 |
| FWM | 3.24×10^{-3} | 1.14×10^{-36} | 1.59×10^{-569} | 15.92 |
| AKM1 | 1.43×10^{-5} | 3.99×10^{-78} | 7.98×10^{-1215} | 15.67 |
| AKM2 | 1.55×10^{-4} | 1.54×10^{-57} | 6.74×10^{-892} | 15.74 |

Table 5. Numerical results of methods without memory for $F_3(x)$.

| $F_3(x) = x^4 + 11.50x^3 + 47.49x^2 + 83.06325x + 51.23266875,$ $x_0 = 1.09, \eta = 1$ | | | | |
|---|-----------------------|-------------------------|-------------------------|------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 1.10×10^{-9} | 3.80×10^{-71} | 7.82×10^{-563} | 7.99 |
| M1 | 4.27×10^{-8} | 4.58×10^{-57} | 8.12×10^{-449} | 7.99 |
| FWM | 4.27×10^{-8} | 4.58×10^{-57} | 8.12×10^{-449} | 7.99 |
| AK1 | 1.64×10^{-8} | 7.60×10^{-61} | 1.63×10^{-479} | 7.99 |
| AK2 | 4.37×10^{-9} | 8.300×10^{-66} | 1.41×10^{-519} | 7.99 |

Table 6. Numerical results of methods with memory for $F_3(x)$.

| $F_3(x) = x^4 + 11.50x^3 + 47.49x^2 + 83.06325x + 51.23266875,$ $x_0 = 1.09, \eta = 1$ | | | | |
|---|-----------------------|-------------------------|--------------------------|-------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 1.10×10^{-9} | 8.99×10^{-143} | 3.61×10^{-1998} | 13.94 |
| M1 | 4.27×10^{-8} | 1.10×10^{-116} | 3.95×10^{-1854} | 15.99 |
| FWM | 4.27×10^{-8} | 1.10×10^{-116} | 3.95×10^{-1854} | 15.99 |
| AKM1 | 1.64×10^{-8} | 2.45×10^{-123} | 1.52×10^{-1960} | 15.99 |
| AKM2 | 4.37×10^{-9} | 1.57×10^{-132} | 3.71×10^{-1998} | 15.11 |

Table 7. Numerical results of methods without memory for $F_4(x)$.

| $F_4(x) = \frac{1}{3x^4} - x^2 - \frac{1}{3x} + 1, x_0 = 1.25, \eta = 1$ | | | | |
|--|-----------------------|------------------------|-------------------------|------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 6.33×10^{-7} | 5.89×10^{-50} | 3.29×10^{-394} | 7.99 |
| M1 | 1.38×10^{-8} | 5.58×10^{-64} | 3.94×10^{507} | 8.00 |
| FWM | 1.38×10^{-8} | 5.58×10^{-64} | 3.94×10^{507} | 8.00 |
| AK1 | 1.38×10^{-8} | 1.22×10^{-63} | 4.54×10^{-504} | 7.99 |
| AK2 | 1.38×10^{-8} | 1.48×10^{-63} | 2.60×10^{-503} | 7.99 |

Table 8. Numerical results of methods with memory for $F_4(x)$.

| $F_4(x) = \frac{1}{3x^4} - x^2 - \frac{1}{3x} + 1, x_0 = 1.25, \eta = 1$ | | | | |
|--|-----------------------|-------------------------|--------------------------|-------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 6.33×10^{-7} | 1.43×10^{-95} | 1.01×10^{-1469} | 15.50 |
| M1 | 1.38×10^{-8} | 1.16×10^{-122} | 2.13×10^{-1891} | 15.50 |
| FWM | 1.38×10^{-8} | 1.16×10^{-122} | 2.13×10^{-1891} | 15.50 |
| AKM1 | 1.38×10^{-8} | 1.19×10^{-122} | 2.93×10^{-1891} | 15.50 |
| AKM2 | 1.38×10^{-8} | 1.20×10^{-122} | 3.44×10^{-1891} | 15.50 |

Table 9. Numerical results of methods without memory for $F_5(x)$.

| $F_5(x) = \sin(x) - \frac{1}{100}x, x_0 = 0.7, \eta = 0$ | | | | |
|--|-----------------------|------------------------|-------------------------|------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 1.60×10^{-5} | 1.24×10^{-43} | 1.56×10^{-348} | 8.00 |
| M1 | 8.83×10^{-7} | 8.08×10^{-54} | 4.01×10^{-430} | 7.99 |
| FWM | 8.83×10^{-7} | 8.08×10^{-54} | 4.01×10^{-430} | 7.99 |
| AK1 | 6.19×10^{-7} | 4.26×10^{-55} | 2.15×10^{-440} | 7.99 |
| AK2 | 4.33×10^{-7} | 2.32×10^{-56} | 1.56×10^{-450} | 7.99 |

Table 10. Numerical results of methods with memory for $F_5(x)$.

| $F_5(x) = \sin(x) - \frac{1}{100}x, x_0 = 0.7, \eta = 0$ | | | | |
|--|-----------------------|-------------------------|--------------------------|-------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 1.60×10^{-5} | 1.23×10^{-80} | 1.20×10^{-1260} | 15.71 |
| M1 | 8.83×10^{-7} | 5.98×10^{-99} | 1.26×10^{-1547} | 15.72 |
| FWM | 8.83×10^{-7} | 5.98×10^{-99} | 1.26×10^{-1547} | 15.72 |
| AKM1 | 6.19×10^{-7} | 6.68×10^{-101} | 3.56×10^{-1578} | 15.72 |
| AKM2 | 4.33×10^{-7} | 6.74×10^{-103} | 1.95×10^{-1609} | 15.72 |

5.1. Location of Maximum Energy Distribution

Planck’s radiation law

$$\theta = \frac{8\pi abq^{-5}}{e^{ab/quv} - 1},$$

where θ is the energy density within an isothermal black body, q is the wavelength of radiation, v is the absolute temperature of the black body, b is Planck’s constant, u is the

Boltzmann’s constant, and a is the speed of light. To determine the wavelength which maximizes the energy density, we calculate

$$\frac{d\theta}{dq} = \frac{8\pi abq^{-5}}{e^{ab/quv} - 1} \left(-5 + \frac{(ab/quv)e^{ab/quv}}{e^{ab/quv} - 1} \right).$$

The term in front of the parentheses is zero in the limits, as $q \rightarrow 0$ and as $q \rightarrow \infty$, although both of these situations give rise to minima in the energy density. The maximum we are seeking arises when the term inside the parenthesis is zero. This happens when

$$1 - \frac{ab}{5q_{\max}uv} = e^{-ab/q_{\max}uv},$$

where q_{\max} is the wavelength that maximizes the energy density. If we let $x = ab/q_{\max}uv$, then the equation for the maximum becomes

$$1 - \frac{x}{5} = e^{-x}.$$

Let us define the function

$$F(x) = e^{-x} - 1 + \frac{x}{5} = 0. \tag{50}$$

The problem is now converted into a root-finding problem, as shown in (50). The solution of the equation is 4.965114232, 0.

5.2. Eigenvalue Problem

One of the most challenging and toughest tasks of linear algebra concerns the eigenvalues of a large square matrix. Another big task is to find the roots of the characteristics equation of a square matrix greater than 4. So, we consider the following 9×9 matrix [19]:

$$S = \frac{1}{8} \begin{pmatrix} -12 & 0 & 0 & 19 & -19 & 76 & -19 & 18 & 437 \\ -64 & 24 & 0 & -24 & 24 & 64 & -8 & 32 & 376 \\ -16 & 0 & 24 & 4 & -4 & 16 & -4 & 8 & 92 \\ -40 & 0 & 0 & -10 & 50 & 40 & 2 & 20 & 242 \\ -4 & 0 & 0 & -1 & 41 & 4 & 1 & 0 & 25 \\ -40 & 0 & 0 & 18 & -18 & 104 & -18 & 20 & 462 \\ -84 & 0 & 0 & -29 & 29 & 84 & 21 & 42 & 501 \\ 16 & 0 & 0 & 4 & 4 & -16 & 4 & 16 & -92 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 24 \end{pmatrix}$$

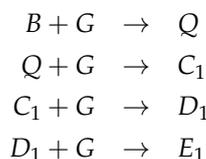
The characteristics polynomial of the above matrix S is given as follows:

$$F(x) = x^9 - 29x^8 + 349x^7 - 2261x^6 + 8455x^5 - 17663x^4 + 15927x^3 + 6993x^2 - 24732x + 12960 \tag{51}$$

The problem is now converted into a root-finding problem, as shown in (51). The solution of the equation is 1, 4, 5, 8, -1, 3, 3, 3, 3.

5.3. Continuous Stirred-Tank Reactor (CSTR)

We assume an isothermal stirred-tank reactor (CSTR). Consider the components B and G , which represent the feed rates to reactors Q and $q - Q$, respectively.



To develop simple feedback control systems, Douglas analyzed this problem in [20]. In the investigation of this system, he presented the following equation for the transfer function of the reactor with a proportional control system:

$$G_c \frac{2.98(x + 2.25)}{(x + 1.45)(x + 2.85)^2(x + 4.35)} = -1,$$

where G_c is the gain of the proportional controller. If we take $G_c = 0$, we obtain the nonlinear equation:

$$F(x) = x^4 + 11.50x^3 + 47.49x^2 + 83.06325x + 51.23266875. \tag{52}$$

Transfer equations have four negative real roots, given as

$$x_1 = -1.45; x_2 = -2.85; x_3 = -2.85; x_4 = -4.45.$$

These roots are called the poles of the open-loop transfer function. Table 11 shows the test functions, exact roots, and initial approximations.

For the sake of comparison, a variety of test functions are considered. Our without-memory methods (AK1) and (AK2) performed better for test functions $F_2(x)$, $F_4(x)$, $F_5(x)$, $F_6(x)$, and $F_7(x)$. In the case of the with-memory methods, (AKM1) and (AKM2) performed exceptionally well for all the test functions $F_1(x)$, $F_2(x)$, $F_3(x)$, $F_4(x)$, $F_5(x)$, $F_6(x)$, and $F_7(x)$. All these results can be verified from Tables 1, 3, 5, 7, 9, 12, and 13 in the case of without-memory methods and from Tables 2, 4, 6, 8, 10, 14, and 15 in the case of with-memory methods. Considering the above-mentioned facts, it is concluded that our newly constructed with- and without-memory methods are robust, proficient, and considerably better than the already existing schemes of similar types.

Table 11. Test functions for comparison of different methods.

| Ex. | Test Functions | Exact Root | Initial Point |
|-----|--|---------------|---------------|
| 1 | $F_1(x) = \exp(-x) - 1 + \frac{1}{5}x$ | $\eta = 0$ | $x_0 = -0.16$ |
| 2 | $F_2(x) = x^9 - 29x^8 + 349x^7 - 2261x^6 + 8455x^5 - 17663x^4 + 15927x^3 + 6993x^2 - 24732x + 12960$ | $\eta = 1$ | $x_0 = 1.09$ |
| 3 | $F_3(x) = x^4 + 11.50x^3 + 47.49x^2 + 83.06325x + 51.23266875$ | $\eta = -4.3$ | $x_0 = -4.35$ |
| 4 | $F_4(x) = \frac{1}{3x^4} - x^2 - \frac{1}{3x} + 1$ | $\eta = 1$ | $x_0 = 1.25$ |
| 5 | $F_5(x) = \sin(x) - \frac{1}{100}x$ | $\eta = 0$ | $x_0 = 0.7$ |
| 6 | $F_6(x) = \arctan(x)$ | $\eta = 0$ | $x_0 = 1$ |
| 7 | $F_7(x) = x^3 + x^2 - 3x - 3$ | $\eta = -1$ | $x_0 = 0.12$ |

Table 12. Numerical results of methods without memory for $F_6(x)$.

| $F_6(x) = \arctan(x), x_0 = 1, \eta = 0$ | | | | |
|--|-----------------------|------------------------|-------------------------|------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 3.90×10^{-2} | 2.16×10^{-16} | 2.14×10^{-130} | 7.99 |
| M1 | 1.98×10^{-2} | 1.36×10^{-19} | 4.56×10^{-156} | 7.95 |
| FWM | 1.98×10^{-2} | 1.36×10^{-19} | 4.56×10^{-156} | 7.95 |
| AK1 | 1.76×10^{-2} | 7.67×10^{-20} | 4.41×10^{-158} | 7.96 |
| AK2 | 4.12×10^{-3} | 2.35×10^{-24} | 3.31×10^{-194} | 7.96 |

Table 13. Numerical results of methods without memory for $F_7(x)$.

| $F_7(x) = x^3 + x^2 - 3x - 3, x_0 = 0.12, \eta = -1$ | | | | |
|--|-----------------------|------------------------|-------------------------|------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 1.60×10^{-2} | 4.92×10^{-15} | 4.66×10^{-115} | 7.99 |
| M1 | 1.74×10^{-4} | 3.61×10^{-29} | 1.27×10^{-226} | 7.99 |
| FWM | 1.74×10^{-4} | 3.61×10^{-29} | 1.27×10^{-226} | 7.99 |
| AK1 | 1.73×10^{-4} | 1.35×10^{-29} | 1.90×10^{-230} | 7.99 |
| AK2 | 1.73×10^{-4} | 6.46×10^{-30} | 2.45×10^{-233} | 7.99 |

Table 14. Numerical results of methods with memory for $F_6(x)$.

| $F_6(x) = \arctan(x), x_0 = 1, \eta = 0$ | | | | |
|--|-----------------------|------------------------|-------------------------|-------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 3.90×10^{-2} | 6.65×10^{-25} | 1.07×10^{-382} | 15.71 |
| M1 | 1.98×10^{-2} | 1.02×10^{-30} | 1.02×10^{-474} | 15.69 |
| FWM | 1.98×10^{-2} | 1.02×10^{-30} | 1.02×10^{-474} | 15.69 |
| AKM1 | 1.76×10^{-2} | 2.31×10^{-31} | 6.29×10^{-485} | 15.70 |
| AKM2 | 4.12×10^{-3} | 1.83×10^{-40} | 2.08×10^{-628} | 15.74 |

Table 15. Numerical results of methods with memory for $F_7(x)$.

| $F_7(x) = x^3 + x^2 - 3x - 3, x_0 = 0.12, \eta = -1$ | | | | |
|--|-----------------------|------------------------|-------------------------|-------|
| | $ x_1 - \eta $ | $ x_2 - \eta $ | $ x_3 - \eta $ | COC |
| LM | 1.60×10^{-2} | 7.05×10^{-31} | 2.31×10^{-484} | 15.99 |
| M1 | 1.74×10^{-4} | 1.71×10^{-61} | 1.38×10^{-973} | 15.99 |
| FWM | 1.74×10^{-4} | 1.71×10^{-61} | 1.38×10^{-973} | 15.99 |
| AKM1 | 1.73×10^{-4} | 1.62×10^{-61} | 5.57×10^{-974} | 15.99 |
| AKM2 | 1.73×10^{-4} | 1.58×10^{-61} | 3.72×10^{-974} | 15.99 |

6. Stereographic Projection of Iterative Method without Memory

In this section, we analyzed the reliability of the iterative scheme through stereographic projection, which is a useful tool to provide us significant information about the convergence and stability of without-memory multipoint iterative schemes. This type of analysis has been performed by Andrew Nicklawsky [21].

Stereographic projection is a mapping between a sphere and a plane. The set of points on the surface of the sphere gives a complete representation in 3D space, the directions being the set of lines from the center of the sphere. It is defined as the unit sphere U^2 in 3D space, and R^3 is the set of points (x, y, z) such that

$$U^2 = \{(x, y, z) \in R^3 | x^2 + y^2 + z^2 = 1\} \subset R^3.$$

For the stereographic projection, the transformation of the Cartesian coordinates is as follows:

$$(x, y, z) = \left(\frac{2X}{1 + X^2 + Y^2}, \frac{2Y}{1 + X^2 + Y^2}, \frac{-1 + X^2 + Y^2}{1 + X^2 + Y^2} \right).$$

The transformation of the Polar coordinates is as follows:

$$(r, \theta, z) = \left(\frac{2R}{1 + R^2}, \Theta, \frac{R^2 - 1}{R^2 + 1} \right).$$

The origin of the above two projections is (0,0): the point (0,0,−1) is on the south pole, the interior of the unit circle is mapped on the southern hemisphere, and the unit circle is mapped on the equator. The point (0,0,1) is on the north pole, which is undefined but can be considered the manifestation of infinity as the points growing closer to it come from points in the plane increasingly distant from the origin. We take three complex polynomials to obtain the stereographic projection of different methods, which are

$$\begin{aligned} r_1(z) &= z^3 - 1. \\ r_2(z) &= z^5 - 1. \\ r_3(z) &= z^6 - 1. \end{aligned}$$

To generate the stereographic projection, we used a maximum of 60 iterations with 200 resolutions implemented in the computer programming language MATLAB R2014a. The roots of each polynomial are marked with different colors. $r_1(z)$ has three roots, so it can be seen from the stereographic images of $r_1(z)$ that three colors appeared, and these three colors reflect three roots, where the black color is used for divergence. The same holds in the case of $r_2(z)$, where five colors can be seen, and in the case of $r_3(z)$, where six colors of roots can be seen from the stereographic projections. If the sequence of root-finding iterative schemes converges in a smaller number of iterations, then the color will be bright, and if it is not converging to any of the roots, then the initial root is allotted with black color. In the following figures, we used a lower value of −2 and an upper value of 2 with an increment of 0.1 on polynomials of 3rd, 5th, and 6th degree. The dynamical behavior of the newly proposed without-memory iterative schemes (AK1) and (AK2) and the known schemes without memory by Zafar et al. [13] (FWM), Cordero et al. [18] (M1), and Lotfi et al. [17] (LM) are seen in Figures 1–6.

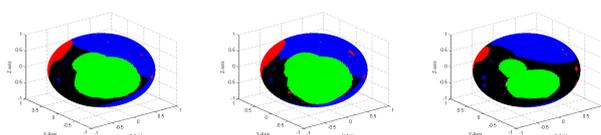


Figure 1. Stereographic projection of AK1 (left), AK2 (middle), and FWM (right) on $r_1(z)$.

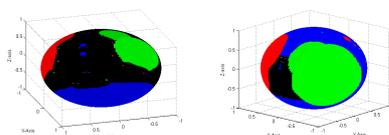


Figure 2. Stereographic projection of M1 (left) and LM (Right) on $r_1(z)$.

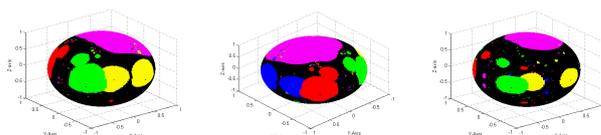


Figure 3. Stereographic projection of AK1 (left), AK2 (middle), and FWM (right) on $r_2(z)$.

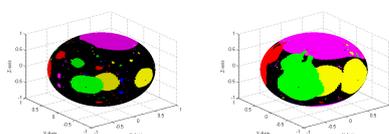


Figure 4. Stereographic projection of M1 (left) and LM (right) on $r_2(z)$.

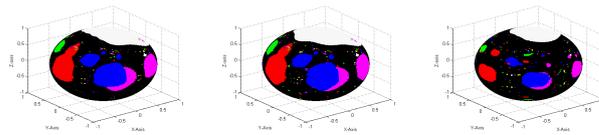


Figure 5. Stereographic projection of AK1 (left), AK2 (middle), and FWM (right) on $r_3(z)$.

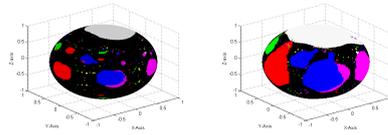


Figure 6. Stereographic projection of M1 (left) and LM (right) on $r_3(z)$.

Figures 1–6 show the region of convergence and divergence through the stereographic projection of the developed families (AK1) and (AK2) compared with the existing families of the same domain. It can be seen from Figures 1 and 2, that the newly developed methods AK1 and AK2 have wider and brighter regions of convergence and less regions of divergence as compared with FWM and M1, and their stability is comparable with LM in the case of $r_1(z)$. The same behavior can be observed from Figures 3 and 4 in the case of polynomial $r_2(z)$ and Figures 5 and 6 in the case of $r_3(z)$. Since the dynamical behavior of the presented families (AK1) and (AK2) have less black color in contrast with (FWM) and (M1), the stability and consistency of the newly proposed family is evident from the stereographic projection.

Without-memory iterative methods are algorithms that determine a subsequent iteration exclusively from the previous one. The current iteration, which is located on the sphere, can be converted to a point on the plane via stereographic projection. The next iterate is then obtained by applying the iteration process in the plane and projecting the resulting point back onto the sphere. This method can make the computation of iterates simpler, because working with points on a plane rather than a sphere may be simpler. Additionally, stereographic projection can show how iterates move on the sphere during the progress of an iteration, which can give geometric insight into how iterative techniques behave.

7. Dynamical Analysis of Iterative Methods With Memory

It is interesting to visualize the convergence region and stability of an iterative scheme through complex tools, such as symmetric basins of attraction and stereographic projections. The basic definitions and complex behavior of rational functions can be found in [22–25]. In the previous section, we investigated stereographic projections, and here, we analyze their symmetric basins of attractions for various iterative methods.

From a numerical viewpoint, complex properties of rational functions associated with iterative schemes give us significant information about their stability. To generate symmetric basins of attractions, we used two different schemes on Matlab R2014a. We consider a square box $B = [-2, 2] \times [-2, 2] \in \mathbb{C}$ with 200×200 grids. An initial approximation for the zero of a polynomial lies in the basins of attractions if an iterative method converges to it with stopping criterion; $|f(\xi_m)| < 10^{-5}$.

The first scheme assigns a specific shade to the initial points in order to visualize the convergence speed as a function of the number of iterations. Each initial point is assigned a unique color. If the iterative scheme's sequence converges in fewer iterations, the color is brighter; if it does not converge to any of the roots after 30 iterations, the initial root is assigned a dark blue color. We use the same scale in the second method but assign a shade to each starting point based on the number of iterations required for the root-finding scheme to converge to any of the given function's roots. We use a maximum of 25 iterations and the same stopping criteria as previously stated. If an iterative scheme does not converge within the specified number of iterations, it is said to be divergent for

that particular starting point and is denoted by black shade. To generate symmetric basins of attractions (polynomiographs), we employ three complex functions, which are

$$\begin{aligned}
 y_3(\zeta) &= \zeta^3 - 1, \zeta = 1.0, -0.5000 + 0.86605I, -0.5000 - 0.86605I \\
 y_5(\zeta) &= \zeta^5 - 1, \zeta = 1.0, 0.3090 + 0.95105I, -0.8090 + 0.58778I, \\
 &\quad -0.8090 - 0.58778I, 0.30902 - 0.95105I. \\
 y_6(\zeta) &= \zeta^6 - \frac{1}{2}\zeta^5 + \frac{11(i+1)}{4}\zeta^4 - \frac{3i+19}{4}\zeta^3 + \frac{5i+11}{4}\zeta^2 + \frac{i-11}{4}\zeta + \frac{3}{2} - 3i, \\
 \zeta &= -1.0068 + 2.0047i, 0.0281 + 0.9963i, 0.0279 - 1.5225i, \\
 &\quad 1.0235 - 0.9556i, 0.9557 - 0.0105i, -0.5284 - 0.5125i.
 \end{aligned}$$

The symmetric basins of attractions of the proposed with-memory schemes (AKM1) and (AKM2) and the existing methods with memory by Zafar et al. [13] (FWM), Lotfi et al. [17] (LM), and Cordero et al. [18] (M1) are shown in Figures 7–9.

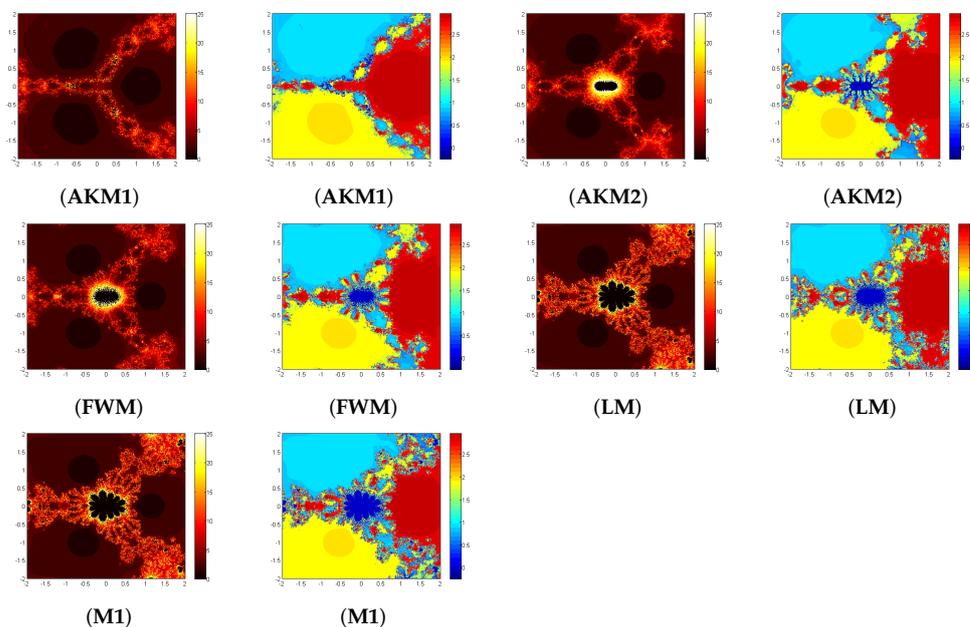


Figure 7. The convergence regions of different methods with memory for $y_3(\zeta)$.

According to Figures 7–9, we conclude that with-memory methods (45) and (46), depending upon self-accelerating parameters calculated by (12), have the fastest order of convergence, as their convergence regions are darker and brighter than the other known schemes. Our methods (AKM1) and (AKM2) have wider convergence regions as compared with (FWM), (LM), and (M1). Thus, the proposed schemes are more reliable and have better convergence regions than (FWM), (LM), and (M1).

Additionally, the investigation of attraction basins can indicate the possibility of different solutions, which could have significant implications for the modeled system. In physics, for instance, the existence of numerous stable solutions can cause phase transitions, while in optimization problems, the existence of several solutions can point to the existence of local minima or maxima. In conclusion, the analysis of attraction basins offers a potential tool for comprehending the behavior of iterative approaches and enhancing their effectiveness. It can help professionals and researchers across a range of disciplines in more effectively and precisely resolving challenging issues.

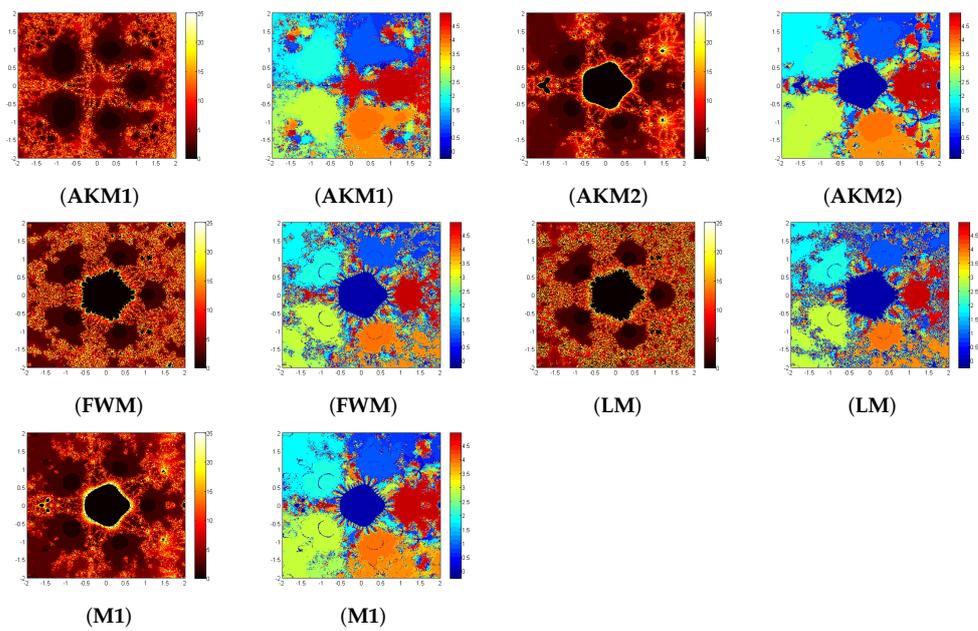


Figure 8. The convergence regions of different methods with memory for $y_5(\zeta)$.

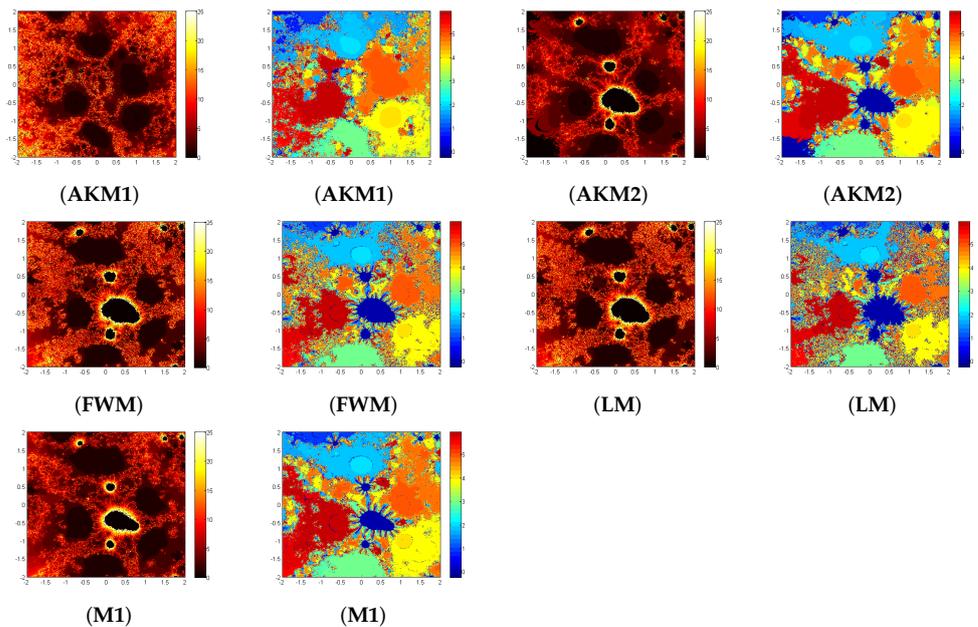


Figure 9. The convergence regions of different methods with memory for $y_6(\zeta)$.

8. Conclusions

In this paper, we developed a new family of optimal eighth-order derivative-free iterative method without memory for finding simple roots of nonlinear equations based on King’s scheme [2] and Lagrange interpolation. A convergence analysis is presented to show that the proposed method without memory has an optimal order of convergence of eight. Furthermore, we used four self-accelerating parameters and one variable’s weight function to extend the special cases (AK1) and (AK2) of the proposed family without memory to iterative schemes with memory, (AKM1) and (AKM2), without using additional function evaluations. We successfully increased the order of convergence from 8 to 15.51560. The efficiency index of the proposed family without memory is increased from 1.6817 to 1.9847 without using additional functional evaluations. For the sake of comparison and implementation of the proposed and existing iterative schemes with and without memory, numerical results are presented using various test problems, including the eigenvalue

problem, continuous stirred-tank reactor, and energy distribution for Planck's radiation. Various dynamical approaches are presented to see the effectiveness of competitive iterative schemes: 3D stereographic projection in the case of iterative schemes without memory and 2D symmetric basins of attractions in the case of iterative schemes with memory. Such a rich analysis is rarely found in the literature. From both approaches, it is evident that our newly developed schemes are more stable, robust, and competent in their respective domains. Future research and analysis of the proposed scheme may be carried out in terms of stability and wide convergence zones in various domains. Visualizing the behavior of the iterative approaches may need looking into the usage of other graphical tools in addition to 2D symmetric basins of attraction and 3D stereographic projections.

Author Contributions: Methodology, S.A. (Saima Akram); Software, S.A. (Saima Akram); Formal analysis, S.A. (Saima Akram) and M.K.; Investigation, M.K.; Resources, S.K.; Data curation, S.K.; Writing—original draft, M.-u.D.J.; Writing—review & editing, M.-u.D.J. and S.A. (Shazia Altaf); Visualization, S.A. (Shazia Altaf); Supervision, S.A. (Saima Akram). All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to express our gratitude to the anonymous reviewers for their comments and suggestions to improve the manuscript.

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

Sample Availability: Not applicable.

References

1. Traub, J.F. *Iterative Methods for the Solution of Equations*; PrenticeHall, Inc.: Englewood Cliffs, NJ, USA, 1964.
2. King, R.F. A family of fourth order methods for non-linear equations. *SIAM J. Numer. Anal.* **1973**, *10*, 876–879. [[CrossRef](#)]
3. Ostrowski, A.M. *Solution of Equations and System of Equations*; Academic Press: New York, NY, USA, 1960.
4. Kung, H.T.; Traub, J.F. Optimal order of one point and multipoint iteration. *J. Assoc. Comput. Math.* **1974**, *21*, 643–651. [[CrossRef](#)]
5. Steffensen, I.F. Remarks on iteration. *Skand. Aktuarietidskr.* **1933**, *16*, 64–72. [[CrossRef](#)]
6. Akram, S.; Zafar, F.; Junjua, M.D.; Yasmin, N. A general family of derivative free with and without memory root finding methods. *JPRM* **2020**, *16*, 64–83.
7. Džunić, J.; Petković, M.S. On generalized multipoint root-solvers with memory. *Comput. Appl. Math.* **2012**, *236*, 2909–2920. [[CrossRef](#)]
8. Junjua, M.D.; Zafar, F.; Yasmin, N.; Akram, S. A general class of derivative free root solvers with-memory. *U.P.B. Sci. Bull. Ser. A* **2017**, *79*, 19–28.
9. Lotfi, T.; Soleymani, F.; Ghorbanzadeh, M.; Assari, P. On the construction of some tri-parametric iterative methods with memory. *Numer. Algorithms* **2015**, *70*, 835–845. [[CrossRef](#)]
10. Petković, M.S.; Neta, B.; Petković, L.D.; Džunić, J. *Multipoint Methods for Solving Nonlinear Equations*; Elsevier: Amsterdam, The Netherlands, 2013.
11. Petković, M.S.; Neta, B.; Petković, L.D.; Džunić, J. Multipoint methods for solving nonlinear equations: A survey. *Appl. Math. Comput.* **2014**, *226*, 635–660. [[CrossRef](#)]
12. Solaiman, O.S.; Ariffin, S.; Karim, A.; Hashim, I. Optimal fourth- and eighth-order of convergence derivative-free modifications of King's method. *J. King Saud Univ.-Sci.* **2019**, *31*, 1499–1504. [[CrossRef](#)]
13. Zafar, F.; Akram, S.; Yasmin, N.; Junjua, M. On the construction of three step four parametric methods with accelerated order of convergence. *J. Nonlinear Sci. Appl.* **2016**, *9*, 4542–4553. [[CrossRef](#)]
14. Zafar, F.; Cordero, A.; Torregrosa, J.R.; Ra, A. A class of four parametric with- and without-memory root finding methods. *Comput. Math. Methods* **2019**, *1*, e1024. [[CrossRef](#)]
15. Zafar, F.; Yasmin, N.; Kutbi, M.A.; Zeshan, M. Construction of tri-parametric derivative free fourth order with and without memory iterative method. *J. Nonlinear Sci. Appl.* **2016**, *9*, 1410–1423. [[CrossRef](#)]
16. Petkovic, M.S.; Zunic, J.; Petkovic, L.D. A family of two point methods with-memory for solving nonlinear equations. *Appl. Anal. Discret. Math.* **2011**, *5*, 298–317. [[CrossRef](#)]
17. Lotfi, T.; Assari, P. New three- and four-parametric iterative with-memory methods with efficiency index near 2. *Appl. Math. Comput.* **2015**, *270*, 1004–1010. [[CrossRef](#)]
18. Cordero, A.; Janjua, M.; Torregrosa, J.R.; Yasmin, N.; Zafar, F. Efficient four parametric with and without-memory iterative methods possessing high efficiency indices. *Math. Prob. Eng.* **2018**, *2018*, 8093673. [[CrossRef](#)]

19. Behl, R.; Zafar, F.; Alshormani, A.S.; Junjua, M.; Yasmin, N. An Optimal Eighth-Order Scheme for Multiple Zeros of Univariate Functions. *Int. J. Comput. Methods* **2019**, *16*, 1843002. [[CrossRef](#)]
20. Douglas, J.M. *Process Dynamics and Control*; Prentice Hall: Englewood Cliffs, NJ, USA, 1972.
21. Nicklawsky, A. Visualizing Chaos. *Mathematics Student Work*. 1. 2012. Available online: [https://digitalcommons.csbsju.edu/math/_\\$students/1](https://digitalcommons.csbsju.edu/math/_$students/1) (accessed on 13 May 2023).
22. Amat, S.; Busquier, S.; Bermúdez, C.; Magrenan, A.A. On the election of the damped parameter of a two-step relaxed Newton-type method. *Nonlinear Dyn.* **2016**, *84*, 9–18. [[CrossRef](#)]
23. Chun, C.; Neta, B. Comparing the basins of attraction for KANwar-Bhatia-Kansal family to the best fourth order method. *Appl. Math. Comput.* **2015**, *266*, 277–292. [[CrossRef](#)]
24. Sharma, J.R.; Kumar, S. Efficient methods of optimal eighth and sixteenth order convergence for solving nonlinear equations. *SeMA* **2018**, *75*, 229–253. [[CrossRef](#)]
25. Neta, B.; Chun, C.; Scott, M. Basins of attraction for optimal eighth order methods to find simple roots of nonlinear equations. *Appl. Math. Comput.* **2014**, *227*, 567–592. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.