

New Derivative-Free Families of Four-Parametric with and Without Memory Iterative Methods for Nonlinear Equations



G Thangkhenpau, Sunil Panday, and Shubham Kumar Mittal

1 Introduction

In the realm of numerical computations, a vast number of problems are expressed using nonlinear equations of the following form:

$$\Omega(s) = 0, \quad (1)$$

where $\Omega : D \subseteq \mathbb{R} \rightarrow \mathbb{R}$ is a real function defined on D , an open interval. Finding the numerical solutions of these problems expressed by (1) has always been a challenging task but at the same time of great importance due to its numerous applications in various branches of science and engineering. Iterative methods are extensively used for solving these problems in order to get approximate solutions of (1) but with high accuracy. The following is one such iterative method, called the Newton's method [1], which is widely used for finding the simple roots of (1):

$$s_{n+1} = s_n - \frac{\Omega(s_n)}{\Omega'(s_n)}, \quad n = 0, 1, 2, \dots \quad (2)$$

It is a classical optimal one-point without memory method with quadratic order of convergence. However, due to the requirement of derivative evaluation and its low convergence order, the Newton's method (2) is not suitable for many practical uses. As a result, various multi-point without memory methods have been developed

G Thangkhenpau (✉) · S. Panday · S. K. Mittal
Department of Mathematics, National Institute of Technology Manipur, Langol, Imphal,
Manipur 795004, India
e-mail: tkpguite92@gmail.com

S. Panday
e-mail: sunilpanday@hotmail.co.in

and studied in literature which have higher convergence order with higher efficiency [2–4]. An iterative method is called optimal if it satisfies the unproved Kung–Traub’s conjecture [5] according to which an iterative without memory method requiring k function evaluations per iteration is optimal if it has the convergence order of 2^{k-1} .

In the past decade, various multi-point with memory methods for finding simple roots of nonlinear equations using accelerating parameters have gained much attention among researchers [6–10]. To theoretically determine the efficiency of an iterative method, Ostrowski [11] introduced the efficiency index $(EI) = p^{\frac{1}{k}}$, where k is the number of function evaluations at each iteration and p is the order of convergence. In fact, it was Traub who first introduces the with memory method, known as the Traub–Steffensen method [1], using a parameter as accelerating parameter. The method is given below:

$$\begin{aligned} w_n &= s_n + \alpha_n \Omega(s_n), \alpha_n \neq 0 \\ s_{n+1} &= s_n - \frac{\Omega(s_n)}{\Omega[s_n, w_n]}, \end{aligned} \quad (3)$$

where $\Omega[s_n, w_n] = \frac{\Omega(s_n) - \Omega(w_n)}{s_n - w_n}$ and α_n is the accelerating parameter calculated as follows:

$$\alpha_n = -\frac{1}{N'_1(s_n)}; N_1 = \Omega(s_n) + (s - s_n)\Omega[s_n, w_n], \quad n \geq 0.$$

The method (3) has convergence order of 2.41 which is higher than the quadratic convergence order of Newton’s method. This is achieved without any additional function evaluation. Also, unlike Newton’s method, Traub–Steffensen method does not require any evaluation of the derivatives and is derivative-free. This has motivated us to develop new multi-point with and without memory iterative methods containing more number of accelerating parameters with increased order of convergence having high efficiency index of almost 2.

In this paper, we introduce new derivative-free four-parametric families of four-point with and without memory iterative methods for computing simple roots of nonlinear equations. Formulation of the family of with memory methods is based on the extension of the new family of without memory methods by using accelerating parameters without any extra function evaluations. As a result, the convergence order increases from 8 to 15.5156. The accelerating parameters are approximated by Newton’s interpolatory polynomials through the best-saved points so as to obtain highly efficient family of with memory methods.

The remaining content of the paper has been structured as follows. In Sect. 2 development of the new derivative-free family of without memory methods is discussed, and the theoretical convergence properties are fully investigated. Section 3 deals with the development and convergence analysis of the new derivative-free family of with memory methods. Section 4 covers the numerical results and the comparison of the proposed families of with and without memory methods with other existing methods

on some test functions. Some real-world problems have been included in this section to confirm the applicability of the proposed families of with and without memory methods. Finally, Sect. 5 presents some concluding remarks.

2 Optimal Four-Parametric Family of Four-Point Without Memory Methods

Let us first consider the following non-optimal three-point Newton steps of eighth-order containing the first-order derivative.

$$\begin{aligned}
 y_n &= s_n - \frac{\Omega(s_n)}{\Omega'(s_n)} \\
 z_n &= y_n - \frac{\Omega(y_n)}{\Omega'(y_n)} \\
 s_{n+1} &= z_n - \frac{\Omega(z_n)}{\Omega'(z_n)}
 \end{aligned} \tag{4}$$

To minimize the number of function evaluations from the above Eq. (4), we first approximate $\Omega'(y_n)$ using the following expression:

$$\Omega'(y_n) \approx \frac{\Omega'(s_n)}{1 + \frac{\Omega(y_n)}{\Omega(s_n)} \frac{\Omega(s_n) + (\lambda - 1)\Omega(y_n)}{\Omega(s_n) - \Omega(y_n)}}, \quad \lambda \in \mathbb{R} \tag{5}$$

Then, we approximate $\Omega'(s_n)$ from the first two steps of the above Eq. (4) as follows:

$$\begin{aligned}
 y_n &= s_n - \frac{\Omega(s_n)}{\Omega[s_n, w_n] + \beta\Omega(w_n)}, \quad w_n = s_n + \alpha\Omega(s_n) \\
 z_n &= y_n - \left(1 + \frac{\Omega(y_n)}{\Omega(s_n)} \frac{\Omega(s_n) + (\lambda - 1)\Omega(y_n)}{\Omega(s_n) - \Omega(y_n)} \right) \frac{\Omega(y_n)}{\Omega[y_n, w_n] + \beta\Omega(w_n)} \\
 s_{n+1} &= z_n - \frac{\Omega(z_n)}{\Omega'(z_n)},
 \end{aligned} \tag{6}$$

where λ is any real parameter and $\alpha, \beta \in \mathbb{R} - \{0\}$.

We want to make this Eq. (6) optimal as well as derivative-free. So, we approximate $\Omega'(z_n)$ in the last step of (6) by the following polynomial:

$$Q(v) = l_0 + l_1(v - z_n) + l_2(v - z_n)^2 + l_3(v - z_n)^3, \tag{7}$$

where l_0, l_1, l_2 and l_3 are some unknowns to be determined by means of the following conditions:

$$Q(s_n) = \Omega(s_n), \quad Q(y_n) = \Omega(y_n), \quad Q(w_n) = \Omega(w_n), \quad Q(z_n) = \Omega(z_n).$$

Now, solving Eq. (7) under the above conditions and simplifying, we obtain the values of l_0, l_1, l_2 and l_3 as follows:

$$l_0 = \Omega(z_n) \tag{8}$$

$$l_3 = \Omega[s_n, y_n, z_n, w_n] \tag{9}$$

$$l_2 = \Omega[y_n, z_n, w_n] - l_3(y_n + w_n - 2z_n) \tag{10}$$

$$l_1 = \Omega[z_n, w_n] - l_2(w_n - z_n) - l_3(w_n - z_n)^2, \tag{11}$$

where $\Omega[x, y, z] = \frac{\Omega[x,y]-\Omega[y,z]}{x-z}$ and $\Omega[x, y, z, v] = \frac{\Omega[x,y,z]-\Omega[y,z,v]}{x-v}$ are second and third divided differences, respectively.

Using (9), (10) and (11), the approximation of $\Omega'(z_n)$ from Eq. (7) is obtained as follows:

$$\Omega'(z_n) \approx Q'(z_n) = l_1 = \Omega[z_n, w_n] - l_2(w_n - z_n) - l_3(w_n - z_n)^2. \tag{12}$$

Now, substituting (12) in (6) and adding two new parameters $\gamma, \delta \in \mathbb{R} - \{0\}$ in the last two steps, we obtain a new optimal derivative-free family of four-point without memory methods which are presented below. We shall denote it by FM8.

$$\begin{aligned} w_n &= s_n + \alpha\Omega(s_n) \\ y_n &= s_n - \frac{\Omega(s_n)}{\Omega[s_n, w_n] + \beta\Omega(w_n)} \\ z_n &= y_n - \left(1 + \frac{\Omega(y_n)}{\Omega(s_n)} \frac{\Omega(s_n) + (\lambda - 1)\Omega(y_n)}{\Omega(s_n) - \Omega(y_n)}\right) \times \\ &\quad \frac{\Omega(y_n)}{\Omega[y_n, w_n] + \beta\Omega(w_n) + \gamma(y_n - s_n)(y_n - w_n)} \\ s_{n+1} &= z_n - \frac{\Omega(z_n)}{Q'(z_n) + \delta(z_n - s_n)(z_n - y_n)(z_n - w_n)} \end{aligned} \tag{13}$$

It is evident that the new family of without memory methods (13) consumes only four function evaluations per full iteration and is completely derivative-free. Also, it preserves the optimal convergence order eighth with the efficiency index $8^{\frac{1}{4}} \approx 1.682$. Now, we present the following theorem through which the convergence criteria of (13) are theoretically discussed.

Theorem 1 *Let $\xi \in D$ be a simple root of a sufficiently differentiable real function $\Omega : D \subseteq \mathbb{R} \rightarrow \mathbb{R}$, where D is an open interval. If an initial guess s_0 is close enough to*

ξ , then the family of proposed methods defined by (13) has eighth-order convergence for any $\lambda \in \mathbb{R}$ and $\alpha, \beta, \gamma, \delta \in \mathbb{R} - \{0\}$. And, it has the error equation given by:

$$\begin{aligned} \varepsilon_{n+1} = & \frac{1}{\Omega'(\xi)^2} (1 + \Omega'(\xi)\alpha)^4 (\beta + d_2)^2 \\ & (-\gamma + \Omega'(\xi)(1 + \Omega'(\xi)\alpha)\beta^2(-1 + \lambda) \\ & + \Omega'(\xi)(2\beta(-2 + \Omega'(\xi)\alpha(-1 + \lambda) + \lambda)d_2 \\ & + (-3 + \Omega'(\xi)\alpha(-1 + \lambda) + \lambda)d_2^2 + d_3)) \\ & (\delta + d_2(-\gamma + \Omega'(\xi)(1 + \Omega'(\xi)\alpha)\beta^2(-1 + \lambda) \\ & + \Omega'(\xi)(2\beta(-2\Omega'(\xi)\alpha(-1 + \lambda) + \lambda)d_2 \\ & + (-3 + \Omega'(\xi)\alpha(-1 + \lambda) + \lambda)d_2^2 + d_3)) \\ & - \Omega'(\xi)d_4)\varepsilon_n^8 + O(\varepsilon_n^9) \end{aligned} \tag{14}$$

where $d_j = \frac{1}{j!} \frac{\Omega^{(j)}(\xi)}{\Omega'(\xi)}$, $j = 2, 3, \dots$, and $\varepsilon_n = s_n - \xi$ is the error at n^{th} iteration.

Proof We construct and apply the following self-explained Mathematica code for the proof of the optimal eighth order of convergence of (13).

$$\Omega[\varepsilon_-] := \Omega'(\xi)(\varepsilon + d_2\varepsilon^2 + d_3\varepsilon^3 + d_4\varepsilon^4 + d_5\varepsilon^5 + d_6\varepsilon^6 + d_7\varepsilon^7 + d_8\varepsilon^8);$$

$$\varepsilon_w = \varepsilon + \alpha\Omega[\varepsilon] \quad (*\varepsilon_w = w - \xi*)$$

$$\text{Out}[1] : (1 + \Omega'(\xi)\alpha)\varepsilon + O[\varepsilon]^2$$

$$\Omega[x_-, y_-] := \frac{\Omega[x] - \Omega[y]}{x - y};$$

$$\Omega[x_-, y_-, z_-] := \frac{\Omega[x, y] - \Omega[y, z]}{x - z};$$

$$\varepsilon_y = \text{Series}\left[\varepsilon - \frac{\Omega[\varepsilon]}{\Omega[\varepsilon, \varepsilon_w] + \beta\Omega[\varepsilon_w]}, \{\varepsilon, 0, 8\}\right] // \text{Simplify}$$

$$\text{Out}[2] : (1 + \Omega'(\xi)\alpha)(\beta + d_2)\varepsilon^2 + O[\varepsilon]^3$$

$$\begin{aligned} \varepsilon_z = & \varepsilon_y - \left(1 + \frac{\Omega[\varepsilon_y]}{\Omega[\varepsilon]} \frac{\Omega[\varepsilon] + (\lambda - 1)\Omega[\varepsilon_y]}{\Omega[\varepsilon] - \Omega[\varepsilon_y]}\right) \times \\ & \frac{\Omega[\varepsilon_y]}{\Omega[\varepsilon_y, \varepsilon_w] + \beta\Omega[\varepsilon_w] + \gamma(\varepsilon_y - \varepsilon)(\varepsilon_y - \varepsilon_w)} // \text{Simplify} \end{aligned}$$

$$\begin{aligned} \text{Out[3]} : & \frac{-1}{\Omega'(\xi)}(1 + \Omega'(\xi)\alpha)^2(\beta + d_2) \\ & (-\Omega'(\xi)\beta^2 - \Omega'(\xi)^2\alpha\beta^2 - \gamma + \Omega'(\xi)\beta^2\lambda \\ & + \Omega'(\xi)^2\alpha\beta^2\lambda + 2\Omega'(\xi)\beta(-2 + \Omega'(\xi)\alpha(-1 + \lambda) + \lambda)d_2 \\ & + \Omega'(\xi)(-3 + \Omega'(\xi)\alpha(-1 + \lambda) + \lambda)d_2^2 + \Omega'(\xi)d_3)\varepsilon^4 + O[\varepsilon]^5 \end{aligned}$$

$$l_3 = \Omega[\varepsilon, \varepsilon_y, \varepsilon_z, \varepsilon_w];$$

$$l_2 = \text{Series}[\Omega[\varepsilon_y, \varepsilon_z, \varepsilon_w] - l_3(\varepsilon_y + \varepsilon_w - 2\varepsilon_z), \{\varepsilon, 0, 8\}]/\text{Simplify};$$

$$Q'(\varepsilon_z) = \text{Series}[\Omega[\varepsilon_z, \varepsilon_w] - l_2(\varepsilon_w - \varepsilon_z) - l_3(\varepsilon_w - \varepsilon_z)^2, \{\varepsilon, 0, 8\}]/\text{Simplify};$$

$$\varepsilon_{n+1} = \text{Series}\left[\varepsilon_z - \frac{\Omega[\varepsilon_z]}{Q'(\varepsilon_z) + \delta(\varepsilon_z - \varepsilon)(\varepsilon_z - \varepsilon_y)(\varepsilon_z - \varepsilon_w)}, \{\varepsilon, 0, 8\}\right]/\text{FullSimplify}$$

$$\begin{aligned} \text{Out[4]} : & \frac{1}{\Omega'(\xi)^2}(1 + \Omega'(\xi)\alpha)^4(\beta + d_2)^2 \\ & (-\gamma + \Omega'(\xi)(1 + \Omega'(\xi)\alpha)\beta^2(-1 + \lambda) + \Omega'(\xi) \\ & (2\beta(-2 + \Omega'(\xi)\alpha(-1 + \lambda) + \lambda)d_2 \\ & + (-3 + \Omega'(\xi)\alpha(-1 + \lambda) + \lambda)d_2^2 + d_3) \\ & (\delta + d_2(-\gamma + \Omega'(\xi)(1 + \Omega'(\xi)\alpha)\beta^2(-1 + \lambda)) - \Omega'(\xi)d_4) \\ & + \Omega'(\xi)(2\beta(-2\Omega'(\xi)\alpha(-1 + \lambda) + \lambda)d_2) \\ & + (-3 + \Omega'(\xi)\alpha(-1 + \lambda) + \lambda)d_2^2 + d_3)\varepsilon_n^8 + O(\varepsilon_n^9) \end{aligned}$$

which shows that Eq. (13) is of optimal order eighth.

3 Four-Parametric Family of Four-Point with Memory Methods

From the error Eq. (14), the convergence order can be increased from 8 to 16 for the method (13) if $\alpha = -\frac{1}{\Omega'(\xi)}$, $\beta = -d_2$, $\gamma = \Omega'(\xi)d_3$ and $\delta = \Omega'(\xi)d_4$. However, the exact value of ξ is not available to us. As such, we shall use $\alpha = \alpha_n$, $\beta = \beta_n$, $\gamma = \gamma_n$ and $\delta = \delta_n$, where α_n , β_n , γ_n and δ_n are accelerating parameters which will be computed using the available information from the current and the previous iterations.

Now, to approximate the accelerating parameters α_n , β_n , γ_n and δ_n , we use interpolation as follows.

$$\alpha_n = -\frac{1}{\mathbb{N}'_4(s_n)}, \beta_n = -\frac{\mathbb{N}''_5(w_n)}{2\mathbb{N}'_5(w_n)}, \gamma_n = \frac{\mathbb{N}'''_6(y_n)}{6}, \delta_n = \frac{\mathbb{N}^{iv}_7(z_n)}{24}, n = 0, 1, 2, \dots \tag{15}$$

where $\mathbb{N}_j(t)$, $j = 4, 5, 6, 7$ are Newton’s interpolatory polynomials of j degrees set through the points, i.e., $s_n, y_n, z_n, w_n, s_{n-1}, y_{n-1}, w_{n-1}, z_{n-1}$.

Now, by using (15) in the method (13), we obtain the following new derivative-free family of with memory methods. We shall denote it by FWM8.

For a given $s_0, \alpha_0, \beta_0, \gamma_0, \delta_0$, we have $w_0 = s_0 + \alpha_0\Omega(s_0)$. Then,

$$\begin{aligned} \alpha_n &= -\frac{1}{\mathbb{N}'_4(s_n)}, \beta_n = -\frac{\mathbb{N}''_5(w_n)}{2\mathbb{N}'_5(w_n)}, \gamma_n = \frac{\mathbb{N}'''_6(y_n)}{6}, \delta_n = \frac{\mathbb{N}^{iv}_7(z_n)}{24} \\ w_n &= s_n + \alpha_n\Omega(s_n) \\ y_n &= s_n - \frac{\Omega(s_n)}{\Omega[s_n, w_n] + \beta_n\Omega(w_n)} \\ z_n &= y_n - \left(1 + \frac{\Omega(y_n)}{\Omega(s_n)} \frac{\Omega(s_n) + (\lambda - 1)\Omega(y_n)}{\Omega(s_n) - \Omega(y_n)}\right) \times \\ &\quad \frac{\Omega(y_n)}{\Omega[y_n, w_n] + \beta_n\Omega(w_n) + \gamma_n(y_n - s_n)(y_n - w_n)} \\ s_{n+1} &= z_n - \frac{\Omega(z_n)}{Q'(z_n) + \delta_n(z_n - s_n)(z_n - y_n)(z_n - w_n)} \end{aligned} \tag{16}$$

Lemma 1 *If $\alpha_n = -\frac{1}{\mathbb{N}'_4(s_n)}$, $\beta_n = -\frac{\mathbb{N}''_5(w_n)}{2\mathbb{N}'_5(w_n)}$, $\gamma_n = \frac{\mathbb{N}'''_6(y_n)}{6}$ and $\delta_n = \frac{\mathbb{N}^{iv}_7(z_n)}{24}$, $n = 0, 1, 2, \dots$, then the following estimates*

$$1 + \alpha_n\Omega'(\alpha) \sim \varepsilon_{n-1}\varepsilon_{n-1,y}\varepsilon_{n-1,w}\varepsilon_{n-1,z} \tag{17}$$

$$\beta_n + d_2 \sim \varepsilon_{n-1}\varepsilon_{n-1,y}\varepsilon_{n-1,w}\varepsilon_{n-1,z} \tag{18}$$

$$K_n \sim \varepsilon_{n-1}\varepsilon_{n-1,y}\varepsilon_{n-1,w}\varepsilon_{n-1,z} \tag{19}$$

$$L_n \sim \varepsilon_{n-1}\varepsilon_{n-1,y}\varepsilon_{n-1,w}\varepsilon_{n-1,z} \tag{20}$$

hold, where

$$\begin{aligned} K_n &= -\gamma_n + \Omega'(\xi)(1 + \Omega'(\xi)\alpha_n)\beta_n^2(-1 + \lambda) \\ &\quad + \Omega'(\xi)(2\beta_n(-2 + \Omega'(\xi)\alpha_n(-1 + \lambda) + \lambda)d_2 \\ &\quad + (-3 + \Omega'(\xi)\alpha_n(-1 + \lambda) + \lambda)d_2^2 + d_3), \end{aligned}$$

$$L_n = \delta_n + d_2(-\gamma_n + \Omega'(\xi)(1 + \Omega'(\xi)\alpha_n)\beta_n^2(-1 + \lambda))$$

$$\begin{aligned}
 &+ \Omega'(\xi)(2\beta_n(-2 + \Omega'(\xi)\alpha_n(-1 + \lambda) + \lambda)d_2 \\
 &+ (-3 + \Omega'(\xi)\alpha_n(-1 + \lambda) + \lambda)d_2^2 + d_3)) - \Omega'(\xi)d_4,
 \end{aligned}$$

$$\varepsilon_n = s_n - \xi, \varepsilon_{n,y} = y_n - \xi, \varepsilon_{n,w} = w_n - \xi, \varepsilon_{n,z} = z_n - \xi.$$

Proof As for the proof, see Lemma 1 of [12].

Now, we present the following theorem for analyzing the R-order of convergence [13] of the derivative-free four-parametric family of four-point with memory methods (16).

Theorem 2 *If an initial guess s_0 is sufficiently close to a root ξ of $\Omega(s) = 0$, the parameters α_n , β_n , γ_n and δ_n are calculated by the expressions in (15), then the R-order of convergence of the methods (16) is at least 15.5156.*

Proof Let the sequence of approximations $\{s_n\}$ produced by (16) converges to the root ξ with the order r . Then, we can write

$$\varepsilon_{n+1} \sim \varepsilon_n^r, \text{ where } \varepsilon_n = s_n - \xi. \tag{21}$$

Then,

$$\varepsilon_n \sim \varepsilon_{n-1}^r \tag{22}$$

Thus,

$$\varepsilon_{n+1} \sim \varepsilon_n^r = (\varepsilon_{n-1}^r)^r = \varepsilon_{n-1}^{r^2} \tag{23}$$

Let the iterative sequences $\{w_n\}$, $\{y_n\}$ and $\{z_n\}$ have orders r_1, r_2 and r_3 , respectively. Then, using (21) and (22) gives

$$\varepsilon_{n,w} \sim \varepsilon_n^{r_1} = (\varepsilon_{n-1}^r)^{r_1} = \varepsilon_{n-1}^{rr_1} \tag{24}$$

$$\varepsilon_{n,y} \sim \varepsilon_n^{r_2} = (\varepsilon_{n-1}^r)^{r_2} = \varepsilon_{n-1}^{rr_2} \tag{25}$$

$$\varepsilon_{n,z} \sim \varepsilon_n^{r_3} = (\varepsilon_{n-1}^r)^{r_3} = \varepsilon_{n-1}^{rr_3} \tag{26}$$

From Theorem 1, we have

$$\varepsilon_{n,w} \sim (1 + \alpha_n \Omega'(\xi))\varepsilon_n \tag{27}$$

$$\varepsilon_{n,y} \sim (1 + \alpha_n \Omega'(\xi))(\beta_n + d_2)\varepsilon_n^2 \tag{28}$$

$$\varepsilon_{n,z} \sim (1 + \alpha_n \Omega'(\xi))^2 (\beta_n + d_2) K_n \varepsilon_n^4 \quad (29)$$

$$\varepsilon_{n+1} \sim (1 + \alpha_n \Omega'(\xi))^4 (\beta_n + d_2)^2 K_n L_n \varepsilon_n^8 \quad (30)$$

Using the above Lemma 1 and (27)–(30), we get

$$\varepsilon_{n,w} \sim (1 + \alpha_n \Omega'(\xi)) \varepsilon_n = \varepsilon_{n-1}^{r+r_1+r_2+r_3+1} \quad (31)$$

$$\varepsilon_{n,y} \sim (1 + \alpha_n \Omega'(\xi)) (\beta_n + d_2) \varepsilon_n^2 = \varepsilon_{n-1}^{2r+2r_1+2r_2+2r_3+2} \quad (32)$$

$$\varepsilon_{n,z} \sim (1 + \alpha_n \Omega'(\xi))^2 (\beta_n + d_2) K_n \varepsilon_n^4 = \varepsilon_{n-1}^{4r+4r_1+4r_2+4r_3+4} \quad (33)$$

$$\varepsilon_{n+1} \sim (1 + \alpha_n \Omega'(\xi))^4 (\beta_n + d_2)^2 K_n L_n \varepsilon_n^8 = \varepsilon_{n-1}^{8r+8r_1+8r_2+8r_3+8} \quad (34)$$

Now, comparing the corresponding powers of ε_{n-1} on the right sides of (24) and (31), (25) and (32), (26) and (33), (23) and (34), we get

$$\begin{aligned} rr_1 - r - r_1 - r_2 - r_3 - 1 &= 0 \\ rr_2 - 2r - 2r_1 - 2r_2 - 2r_3 - 2 &= 0 \\ rr_3 - 4r - 4r_1 - 4r_2 - 4r_3 - 4 &= 0 \\ r^2 - 8r - 8r_1 - 8r_2 - 8r_3 - 8 &= 0 \end{aligned} \quad (35)$$

This system has the non-trivial solution $r_1 = 1.9394$, $r_2 = 3.8789$, $r_3 = 7.7578$ and $r = 15.5156$. Hence, the R-order of convergence of the proposed family of methods (16) is at least 15.5156. The proof is complete.

4 Numerical Results

In this section, we examine the performance and the computational efficiency of the newly developed with and without memory methods discussed in Sects. 2 and 3 and compare with some methods of similar nature available in literature. In particular, we have considered for the comparison, the following four-parametric methods: LAM8(3.31) [8], ZM8 (ZR1 from [9]) and ACM8 (M1 from [10]). All numerical tests are executed using the programming software Mathematica 12.2. Throughout the whole computation, we have chosen the same values of the parameters $\alpha_0 = \beta_0 = \gamma_0 = \delta_0 = -1$ and $\lambda = 2$ in all the test functions in order to start the initial iteration. These same values are used for the corresponding parameters of all the compared methods in order to have fair comparison. Numerical test functions which comprise

some standard academic examples and real-life chemical engineering problems along with their simple roots (ξ) and initial guesses (s_0) are presented below.

Example 1 A standard academic test function given by:

$$\Omega_1(s) = e^{-s^2} (1 + s^3 + s^6)(s - 2). \quad (36)$$

It has a simple root $\xi = 2$. We start with the initial guess $s_0 = 2.3$.

Example 2 A standard academic test function given by:

$$\Omega_2(s) = \log(s^2 + s + 2) - s + 1. \quad (37)$$

It has a simple root $\xi \approx 4.1525907367571583$. We start with the initial guess $s_0 = 4.5$.

Example 3 A standard academic test function given by

$$\Omega_3(s) = \sin^2 s + s \quad (38)$$

It has a simple root $\xi = 0$. We start with the initial guess $s_0 = 0.6$.

Example 4 The azeotropic point of a binary solution problem given by the following nonlinear equation (for details see [14]).

$$\Omega_4(s) = \frac{FG(G(1-s)^2 - Fs^2)}{(s(F-G) + G)^2} + 0.14845, \quad (39)$$

where $F = 0.38969$ and $G = 0.55954$.

It has a simple root $\xi \approx 0.69147373574714142$. We start with the initial guess $s_0 = 1.1$.

In Tables 1 and 2, we have displayed the absolute residual errors $|\Omega(s_n)|$ at the first three iterations obtained by the compared methods. We also include the computational convergence order (COC) of each compared method which is computed by the following formula [15]:

$$\text{COC} = \frac{\log|\Omega(s_n)/\Omega(s_{n-1})|}{\log|\Omega(s_{n-1})/\Omega(s_{n-2})|}. \quad (40)$$

From the two Tables 1 and 2, the numerical results affirm the robust performance and high efficiency of the proposed with and without memory methods thus reaffirming their theoretical results. The proposed methods give better accuracy with high efficiency in terms of minimal residual errors after three iterations as compared to the other methods. Further, the COC supports the theoretical convergence order of the new proposed with and without memory methods in the test functions.

Table 1 Comparison of the without memory methods on the test functions

Methods	$\Omega(s)$	$ \Omega(s_1) $	$ \Omega(s_2) $	$ \Omega(s_3) $	COC
ZM8	$\Omega_1(s)$	0.00083523	8.4618×10^{-26}	9.4210×10^{-202}	8.0000
ACM8	$\Omega_1(s)$	0.0061646	1.3517×10^{-20}	7.6376×10^{-162}	7.9986
LAM8	$\Omega_1(s)$	0.000015809	7.9235×10^{-41}	3.1542×10^{-323}	8.0000
FM8	$\Omega_1(s)$	9.6344×10^{-7}	2.6067×10^{-49}	7.4864×10^{-390}	8.0000
ZM8	$\Omega_2(s)$	0.19523	0.076326	3.9752×10^{-6}	10.502
ACM8	$\Omega_2(s)$	0.90214	0.13906	0.000042430	4.3292
LAM8	$\Omega_2(s)$	0.014457	8.6044×10^{-13}	1.5302×10^{-94}	7.9948
FM8	$\Omega_2(s)$	0.011023	7.2266×10^{-15}	3.4512×10^{-112}	7.9880
ZM8	$\Omega_3(s)$	6.7546×10^{-7}	2.7816×10^{-100}	1.9034×10^{-1594}	16.000
ACM8	$\Omega_3(s)$	5.8728×10^{-7}	2.9666×10^{-101}	5.3325×10^{-1610}	16.000
LAM8	$\Omega_3(s)$	1.4810×10^{-6}	7.9355×10^{-95}	3.6632×10^{-1507}	16.000
FM8	$\Omega_3(s)$	1.9346×10^{-7}	5.7032×10^{-109}	1.8559×10^{-1733}	16.000
ZM8	$\Omega_4(s)$	0.074840	1.3522×10^{-10}	1.4582×10^{-80}	8.002
ACM8	$\Omega_4(s)$	Divergent	Divergent	Divergent	—
LAM8	$\Omega_4(s)$	0.012310	1.9635×10^{-16}	8.0818×10^{-127}	8.000
FM8	$\Omega_4(s)$	0.0069383	2.2151×10^{-18}	2.3818×10^{-142}	8.001

Table 2 Comparison of the with memory methods on the test functions

Methods	$\Omega(s)$	$ \Omega(s_1) $	$ \Omega(s_2) $	$ \Omega(s_3) $	COC
ZM8	$\Omega_1(s)$	0.00083523	4.9114×10^{-50}	4.7977×10^{-749}	15.120
ACM8	$\Omega_1(s)$	0.0061646	2.1885×10^{-39}	1.3451×10^{-598}	15.342
LAM8	$\Omega_1(s)$	0.000015809	4.4940×10^{-80}	5.0217×10^{-1237}	15.520
FWM8	$\Omega_1(s)$	9.6344×10^{-7}	1.3319×10^{-96}	5.0044×10^{-1498}	15.596
ZM8	$\Omega_2(s)$	0.19523	4.8315×10^{-25}	3.1255×10^{-375}	14.834
ACM8	$\Omega_2(s)$	0.90214	3.6884×10^{-11}	9.1383×10^{-174}	15.653
LAM8	$\Omega_2(s)$	0.014457	5.3706×10^{-43}	8.9914×10^{-669}	15.478
FWM8	$\Omega_2(s)$	0.011023	1.1528×10^{-44}	3.5410×10^{-695}	15.496
ZM8	$\Omega_3(s)$	6.7546×10^{-7}	1.3098×10^{-95}	5.0428×10^{-1467}	15.459
ACM8	$\Omega_3(s)$	5.872×10^{-7}	8.5805×10^{-99}	3.2362×10^{-1551}	15.816
LAM8	$\Omega_3(s)$	1.4810×10^{-6}	2.2038×10^{-95}	1.5131×10^{-1494}	15.751
FWM8	$\Omega_3(s)$	1.9346×10^{-7}	7.6149×10^{-104}	3.6249×10^{-1632}	15.853
ZM8	$\Omega_4(s)$	0.074840	4.6224×10^{-22}	3.3998×10^{-330}	15.247
ACM8	$\Omega_4(s)$	Divergent	Divergent	Divergent	—
LAM8	$\Omega_4(s)$	0.012310	4.0863×10^{-37}	3.1559×10^{-569}	15.433
FWM8	$\Omega_4(s)$	0.0069383	1.3673×10^{-39}	2.4610×10^{-609}	15.522

5 Concluding Remarks

We have presented in this paper new derivative-free families of with and without memory methods for finding the solutions of nonlinear equations. The use of four accelerating parameters in the with memory methods has enabled us to increase the convergence order of the without memory methods from 8 to 15.5156 and obtain very high-efficiency index of $15.5156^{\frac{1}{4}} \approx 1.9847$ without extra function evaluations. The numerical results further confirm the good performance, validity and applicability of the proposed with and without memory methods. They are found to be more efficient with better accuracy as compared to the existing methods in comparison in terms of minimal residual errors after three iterations for convergence toward the required simple roots.

References

1. Traub JF (1982) Iterative methods for the solution of equations. *Am Math Soc* 312
2. Singh A, Jaiswal JP (2016) A class of optimal eighth-order Steffensen-type iterative methods for solving nonlinear equations and their basins of attraction. *Appl Math Inf Sci* 10(1):251–257. <https://doi.org/10.18576/amis/100125>
3. Panday S, Sharma A, Thangkhenpau G (2023) Optimal fourth and eighth-order iterative methods for non-linear equations. *J Appl Math Comput* 69(1):953–971. <https://doi.org/10.1007/s12190-022-01775-2>
4. Behl R, Alshomrani AS, Chun C (2020) A general class of optimal eighth-order derivative free methods for nonlinear equations. *J Math Chem* 58:854–867
5. Kung HT, Traub JF (1974) Optimal order of one-point and multipoint iteration. *J ACM (JACM)* 21(4):643–651. <https://doi.org/10.1145/321850.321860>
6. Zafara F, Yasmina N, Kutbib MA, Zeshana M (2016) Construction of tri-parametric derivative free fourth order with and without memory iterative method. *J Nonlinear Sci Appl* 9:1410–1423
7. Chanu WH, Panday S, Thangkhenpau G (2022) Development of optimal iterative methods with their applications and basins of attraction. *Symmetry* 14(10):2020. <https://doi.org/10.3390/sym14102020>
8. Lotfi T, Assari P (2015) Two new three and four parametric with memory methods for solving nonlinear equations. *Int J Ind Math* 7(3):269–276
9. Zafar F, Cordero A, Torregrosa JR, Rafi A (2019) A class of four parametric with-and without-memory root finding methods. *Comp and Math Methods* 1:e1024
10. Cordero A, Janjua M, Torregrosa JR, Yasmin N, Zafar F (2018) Efficient four parametric with and without-memory iterative methods possessing high efficiency indices. *Math Prob Eng* 2018:1–12. <https://doi.org/10.1155/2018/8093673>
11. Ostrowski AM (1966) *Solution of equations and systems of equations*. Academic Press, New York-London
12. Džunić J (2013) On efficient two-parameter methods for solving nonlinear equations. *Numer Algor* 63:549–569. <https://doi.org/10.1007/s11075-012-9641-3>
13. Ortega JM, Rheinboldt WG (1970) *Iterative solution of nonlinear equations in several variables*. Academic Press, New York
14. Solaiman OS, Hashim I (2019) Efficacy of optimal methods for nonlinear equations with chemical engineering applications. *Math Probl Eng* 2019, Article ID 1728965, 11 p. <https://doi.org/10.1155/2019/1728965>
15. Petković MS (2011) Remarks on ““On a general class of multipoint root-finding methods of high computational efficiency.”” *SIAM J Numer Math* 49:1317–1319