



# Article Novel Parametric Families of with and without Memory Iterative Methods for Multiple Roots of Nonlinear Equations

G Thangkhenpau<sup>1,\*</sup>, Sunil Panday<sup>1,\*</sup>, Shubham Kumar Mittal<sup>1</sup>, and Lorentz Jäntschi<sup>2</sup>

- <sup>1</sup> Department of Mathematics, National Institute of Technology Manipur, Langol, Imphal 795004, Manipur, India; shubhammehara2@gmail.com
- <sup>2</sup> Department of Physics and Chemistry, Technical University of Cluj-Napoca, B.-dul Muncii nr. 103-105, 400641 Cluj-Napoca, Romania; lorentz.jantschi@gmail.com
- \* Correspondence: tkpguite92@gmail.com (G.T.); sunilpanday@hotmail.co.in (S.P.)

**Abstract:** The methods that use memory using accelerating parameters for computing multiple roots are almost non-existent in the literature. Furthermore, the only paper available in this direction showed an increase in the order of convergence of 0.5 from the without memory to the with memory extension. In this paper, we introduce a new fifth-order without memory method, which we subsequently extend to two higher-order with memory methods using a self-accelerating parameter. The proposed with memory methods extension demonstrate a significant improvement in the order of convergence from 5 to 7, making this the first paper to achieve at least a 2-order improvement. In addition to this improvement, our paper is also the first to use Hermite interpolating polynomials to approximate the accelerating parameter in the proposed with memory methods for multiple roots. We also provide rigorous theoretical proofs of convergence theorems to establish the order of the proposed methods. Finally, we demonstrate the potential impact of the proposed methods through numerical experimentation on a diverse range of problems. Overall, we believe that our proposed methods have significant potential for various applications in science and engineering.

**Keywords:** multiple roots; nonlinear equation; with memory methods; Hermite interpolating polynomial

MSC: 65H05; 41A25; 65D99; 28A80

# 1. Introduction

The quest for efficient and accurate methods for finding the roots of nonlinear equations is a continuous endeavour in the field of numerical computation. One of the primary objectives in this pursuit is to develop methods that are both efficient and have a simple structure. Iterative methods are favoured for finding the roots of nonlinear equations because they can be easily implemented using computer software and provide results up to a desired accuracy, despite not always yielding exact roots. As a result, they are practical and versatile tools for solving nonlinear equations of the form:

$$\Omega(s_n) = 0,\tag{1}$$

where  $\Omega : D \subseteq \mathbb{C} \to \mathbb{C}$  is a function defined in an open interval *D* having multiple roots  $\xi$  with multiplicity m > 1. These equations are often encountered in various fields of science, engineering, and mathematics, and their solutions are important for understanding complex systems and phenomena. The modified Newton–Raphson method is a widely known iterative method for finding multiple roots of (1), and it is given by

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



Citation: Thangkhenpau, G.; Panday, S.; Mittal, S.K.; Jäntschi, L. Novel Parametric Families of with and without Memory Iterative Methods for Multiple Roots of Nonlinear Equations. *Mathematics* **2023**, *11*, 2036. https://doi.org/10.3390/ math11092036

Academic Editors: Theodore E. Simos and Charampos Tsitouras

Received: 3 April 2023 Revised: 20 April 2023 Accepted: 21 April 2023 Published: 25 April 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/). Equation (2) has a quadratic order of convergence for  $m \ge 1$ . While there exist numerous with and without memory iterative methods for computing simple roots [1–6], the task of developing efficient methods for finding multiple roots remains challenging. Despite the availability of several without memory iterative methods for multiple roots (see [7–10] and the references therein), there is a paucity of methods that utilise multiple points with memory and accelerating parameters for computing multiple roots.

The authors in [11] recently extended existing optimal-order without memory methods [12,13] for multiple roots to include memory using an accelerating parameter. The with memory extension of the optimal fourth-order method is given below:

$$y_n = s_n - m \frac{\Omega(s_n)}{\Omega'(s_n) + \alpha_n \Omega(s_n)},$$
  

$$s_{n+1} = y_n - m u_n W(u_n) \frac{\Omega(s_n)}{\Omega'(s_n) + 2\alpha_n \Omega(s_n)},$$
(3)

where  $u_n = \left(\frac{\Omega(y_n)}{\Omega(s_n)}\right)^{\frac{1}{m}}$ . This method (3) has the error equations given by

S

$$e_{n,y} = \frac{(\alpha_n + c_1)}{m} e_n^2 + O(e_n^3),$$
(4)

$$e_{n+1} = -(\alpha_n + c_1) \frac{A}{2m^3} e_n^4 + O(e_n^5),$$
(5)

where  $c_i = \frac{m!}{(m+i)!} \frac{\Omega^{(m+i)}(\xi)}{\Omega^{(m)}(\xi)}$ ,  $i = 1, 2, ..., e_n = s_n - \xi$  is the error at the  $n^{th}$  iteration,  $A = \alpha_n^2(W''(0) - 4) + 2\alpha_n c_1(W''(0) - 7) + (W''(0) - 9 - m)c_1^2 + 2mc_2$ . The accelerating parameter  $\alpha_n$  is calculated as follows:

$$\alpha_n = -c_1 = -\frac{1}{m+1} \frac{N_2^{(m+1)}(s_n)}{\Omega^{(m)}(s_n)},\tag{6}$$

where  $N_2^{(m+1)}(s_n)$  is calculated using a Newton interpolating polynomial.

However, the R-order of convergence of these methods increases from 2 to 2.4142, 4 to 4.2361, and 4 to 4.5616 only, which is a modest improvement. As far as we know, this is the first and only paper that to discuss the with memory iterative method for finding multiple roots using an accelerating parameter.

Motivated by the approach using the self-accelerating technique, we plan to develop new higher-order with memory iterative methods for finding multiple roots of nonlinear equations using an accelerating parameter.

This paper presents new parametric families of two-point with and without memory iterative methods for finding multiple roots of nonlinear equations. The family of without memory methods employs the weight function technique with a real parameter to achieve the fifth order of convergence, while the extended family of with memory methods utilises an accelerating parameter to achieve a two-order improvement in the convergence order. The with memory methods demonstrate an increase in convergence orders from 5 to 7 and 7.2749. To ensure efficiency, we approximated the accelerating parameter using Hermite interpolating polynomials.

The manuscript is structured as follows. Section 2 outlines the development of methods utilising both weight function techniques and the accelerating parameter, with particular attention paid to analysing the order of convergence of the new methods through theorems and lemmas. In Section 3, we present the results of numerical tests comparing the proposed methods with other known methods. Finally, Section 4 provides concluding remarks on the study.

#### 2. Construction of New Iterative Schemes and Their Convergence Analysis

In this section, we present the new parametric families of two-point with and without memory methods for finding multiple roots having multiplicity m > 1 in two separate subsections.

### 2.1. Parametric Family of Two-Point without Memory Methods and Its Convergence Analysis

Here, we introduce a new fifth-order parametric family of two-point without memory methods for finding multiple roots with multiplicity m > 1. The newly proposed parametric family of without memory methods is defined as follows, which we denote as PFM:

$$y_n = s_n - m \frac{\Omega(s_n)}{\Omega'(s_n) + \alpha \Omega(s_n)}, \ \alpha \in \mathbb{R} - \{0\}$$
$$s_{n+1} = y_n - m W(t_n) \frac{\Omega(y_n)}{\Omega'(y_n) + \alpha \Omega(y_n)},$$
(7)

where  $W : \mathbb{C} \to \mathbb{C}$  is an analytic function in the neighbourhood of 0 with  $t_n = \left(\frac{\Omega'(y_n)}{\Omega'(s_n)}\right)^{\frac{1}{m-1}}$ . This new family (7) requires two functions and two first derivative evaluations at each iteration.

**Theorem 1.** Let  $s = \xi$  be a multiple zero with multiplicity m > 1 of a function  $\Omega: \mathbb{C} \to \mathbb{C}$  in the region enclosing  $\xi$ . Then, the new parametric family of methods (PFM) defined by (7) has the fifth order of convergence if the following conditions are fulfilled:

$$W(0) = 1; W'(0) = 0; W''(0) = 2; |W^{(3)}(0)| < \infty.$$
 (8)

*This satisfies the following error equation:* 

$$e_{n+1} = \frac{(\alpha + c_1)^2}{6(m-1)m^4} \Big( (1-m)(12mc_2 + \alpha^2(-6 + W^{(3)}(0))) + 2\alpha c_1(m(12 - W^{(3)}(0))) \\ + W^{(3)}(0)) + c_1^2(6 + 6m^2 - m(-12 + W^{(3)}(0) + W^{(3)}(0))) \Big) e_n^5 + O(e_n^6),$$
(9)

where  $e_n = s_n - \xi$  is the error at the  $n^{th}$  iteration.

**Proof of Theorem 1.** Let  $s = \xi$  be the multiple roots of  $\Omega(s) = 0$  with multiplicity m > 1 such that  $e_n = s_n - \xi$  is the error at the  $n^{th}$  iteration. We expand  $\Omega(s_n)$  and  $\Omega'(s_n)$  in powers of  $e_n$  by Taylor's series expansion as follows:

$$\Omega(s_n) = \frac{\Omega^{(m)}(\xi)}{m!} e_n^m \Big( 1 + c_1 e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + c_5 e_n^5 + O(e_n^6) \Big), \tag{10}$$

and 
$$\Omega'(s_n) = \frac{\Omega^{(m)}(\xi)}{m!} e_n^{m-1} \Big( m + (m+1)c_1 e_n + (m+2)c_2 e_n^2 + (m+3)c_3 e_n^3 + (m+4)c_4 e_n^4 + (m+5)c_5 e_n^5 + O(e_n^6) \Big),$$
 (11)

where  $c_i = \frac{m!}{(m+i)!} \frac{\Omega^{(m+i)}(\xi)}{\Omega^{(m)}(\xi)}$ , i = 1, 2, 3, ... Using the above Equations (10) and (11) in the first step of Equation (7), we obtain

$$y_n - \xi = \frac{\alpha + c_1}{m} e_n^2 - \frac{1}{m^2} \left( \alpha^2 + 2\alpha c_1 + (1+m)c_1^2 - 2mc_2 \right) e_n^3 + \sum_{i=0}^1 S_i e_n^{i+4} + O(e_n^6), \quad (12)$$

where  $S_i, i = 0, 1$  are given in terms of  $\alpha, m, c_1, c_2, c_3, c_4$ , i.e.,  $S_0 = \frac{1}{m^3} (\alpha^3 + (3+2m)\alpha c_1^2 + (1+m)^2 c_1^3 - 4m\alpha c_2 + c_1(3\alpha^2 - m(4+3m)c_2) + 3m^2 c_3), S_1 = -\frac{1}{m^4} (\alpha^4 + 2(2+3m+m^2)\alpha c_1^3 + (1+m)^3 c_1^4 - 6m\alpha^2 c_2 + 2m^2(2+m)c_2^2 - c_1^2(-3(2+m)\alpha^2 + 2m(3+5m+2m^2)c_2) + 6m^2\alpha c_3 + 2c_1(2\alpha^3 - 3m(2+m)\alpha c_2 + m^2(3+2m)c_3) - 4m^3 c_4).$ 

Then, using Equation (12),  $\Omega(y_n)$  and  $\Omega'(y_n)$  can be obtained as follows:

$$\Omega(y_n) = \frac{\Omega^{(m)}(\xi)}{m!} e_n^{2m} \left(\frac{\alpha + c_1}{m}\right)^m \left[1 - \frac{1}{\alpha + c_1} \left(\alpha^2 + 2\alpha c_1 + (1+m)c_1^2 - 2mc_2\right)e_n + \sum_{i=0}^3 K_i e_n^{i+2} + O(e_n^6)\right],$$
(13)

where  $K_i$ , i = 0, 1, 2, 3 are given in terms of  $\alpha$ , m,  $c_1$ ,  $c_2$ ,  $c_3$ ,  $c_4$ , i.e.,

$$K_0 = \frac{1}{2m(\alpha+c_1)^2} \left( 2c_1(\alpha+c_1)^3 + (-1+m)(\alpha^2+2\alpha c_1+(1+m)c_1^2-2mc_2)^2 + 2(\alpha+c_1)(\alpha^3+(3+2m)\alpha c_1^2 + (1+m)^2 c_1^3 - 4m\alpha c_2 + c_1(3\alpha^2-m(4+3m)c_2) + 3m^2 c_3) \right), \text{ etc.}$$
  
In a similar manner,  $\Omega'(y_n)$  can be written as follows:

$$\Omega'(y_n) = \frac{\Omega^{(m)}(\xi)}{(m-1)!} \left(\frac{\alpha+c_1}{m}\right)^{m-1} e_n^{2(m-1)} \left[1 - \frac{(m-1)}{m(\alpha+c_1)} (\alpha^2 + 2\alpha c_1 + (1+m)c_1^2 - 2mc_2)e_n + \sum_{i=0}^3 M_i e_n^{i+2} + O(e_n^6)\right],$$
(14)

where  $M_i$ , i = 0, 1, 2, 3 are given in terms of  $\alpha$ , m,  $c_1$ ,  $c_2$ ,  $c_3$ ,  $c_4$ . Applying Equations (11) and (14), we obtain

$$t_n = \left(\frac{\Omega'(y_n)}{\Omega'(s_n)}\right)^{\frac{1}{m-1}} = \frac{\alpha + c_1}{m}e_n + \frac{1}{(m-1)m^2}\left((\alpha - 3m\alpha)c_1 - m(1+m)c_1^2 + (m-1)(-\alpha^2 + 2mc_2)\right)e_n^2 + \sum_{i=0}^2 N_ie_n^{i+3} + O(e_n^6), \quad (15)$$

where  $N_i$ , i = 0, 1, 2 are given in terms of  $\alpha$ , m,  $c_1$ ,  $c_2$ ,  $c_3$ ,  $c_4$ .

Expanding the weight function  $W(t_n)$  in the neighbourhood of origin by Taylor's series expansion up to second-order terms gives

$$W(t_n) \approx W(0) + t_n W'(0) + \frac{t_n^2}{2!} W''(0).$$
 (16)

Now, substituting the expressions (12)–(16) in the last step of Equation (7), the error equation is obtained as follows:

$$e_{n+1} = \frac{1}{m} (1 - W(0))(\alpha + c_1)e_n^2 + \frac{1}{m^2} \left( 2m(1 - W(0))c_2 + \alpha^2(-1 + W(0) - W'(0)) + 2\alpha c_1(-1 + W(0) - W'(0)) + c_1^2(-1 + m(-1 + W(0)) + W(0) - W'(0)) \right) e_n^3 + \dots + O(e_n^6).$$
(17)

Now, putting the conditions W(0) = 1, W'(0) = 0, and W''(0) = 2 in (17), the error equation becomes

$$e_{n+1} = \frac{(\alpha + c_1)^2}{6(m-1)m^4} \Big( (1-m)(12mc_2 + \alpha^2(-6 + W^{(3)}(0))) + 2\alpha c_1(m(12 - W^{(3)}(0))) \\ + W^{(3)}(0)) + c_1^2(6 + 6m^2 - m(-12 + W^{(3)}(0) + W^{(3)}(0))) \Big) e_n^5 + O(e_n^6).$$

From the above error equation, we can conclude that the newly proposed family of iterative methods (PFM) is of fifth order. This completes the proof.  $\Box$ 

Some Particular Cases of the Weight Function,  $W(t_n)$ :

Based on the conditions on  $W(t_n)$  as shown in Theorem 1, we can generate numerous methods of the family (7). However, we restricted this to the following simple forms:

**Case 1.** Let  $W(t_n)$  be a polynomial function of degree two of the form:

$$W(t_n) = a_0 + a_1 t_n + a_2 t_n^2.$$

Then, using the conditions (8) of Theorem 1, we obtain  $a_0 = a_2 = 1$  and  $a_1 = 0$ . Thus,  $W(t_n)$  becomes

$$W(t_n) = 1 + t_n^2. (18)$$

**Case 2.** Consider  $W(t_n)$  as a rational function of the form:

$$W(t_n) = \frac{1 + a_0 t_n}{1 + a_1 t_n + a_2 t_n^2}.$$

Then, using the conditions (8) of Theorem 1, we obtain  $a_0 = a_1 = \lambda$  and  $a_2 = -1$ . Thus,  $W(t_n)$  becomes

$$W(t_n) = \frac{1 + \lambda t_n}{1 + \lambda t_n - t_n^2}, \ \lambda \in \mathbb{R}.$$
(19)

**Case 3.** Let  $W(t_n)$  be another rational function of the form:

$$W(t_n) = \frac{1 + a_0 t_n^2}{1 + a_1 t_n^2}$$

Then, using the conditions (8) of Theorem 1, we obtain  $a_0 = 2$  and  $a_1 = 1$ . Thus,  $W(t_n)$  becomes

$$W(t_n) = \frac{1 + 2t_n^2}{1 + t_n^2}.$$
(20)

**Case 4.** Let  $W(t_n)$  be a function of the form:

$$W(t_n) = (1 - t_n)e^{a_0 t_n} + a_1 t_n^2.$$

Then, using the conditions (8) of Theorem 1, we obtain  $a_0 = 1$  and  $a_1 = \frac{3}{2}$ . Thus,  $W(t_n)$  becomes

$$W(t_n) = (1 - t_n)e^{t_n} + \frac{3}{2}t_n^2.$$
(21)

### 2.2. Parametric Families of Two-Point with Memory Methods and Their Convergence Analysis

Here, we propose new parametric families of two-point with memory methods, which are an extension of the new fifth-order parametric family of without memory methods (PFM) (7).

By analysing the error Equation (9) of Theorem 1, we found that the convergence order of PFM (7) can be increased from 5 to 7 if we set  $\alpha = -c_1$ , where  $c_1 = \frac{m!}{(m+1)!} \frac{\Omega^{(m+1)}(\xi)}{\Omega^{(m)}(\xi)}$ , m > 1. However, since the exact values of  $\Omega^{(m+1)}(\xi)$  and  $\Omega^{(m)}(\xi)$  are not available, we used some approximations and replaced  $\alpha$  by  $\alpha_n$ , where  $\alpha_n$  is an accelerating parameter computed using available information from the current and previous iterations to satisfy the following condition:

$$\lim_{n \to \infty} \alpha_n = -\frac{m!}{(m+1)!} \frac{\Omega^{(m+1)}(\xi)}{\Omega^{(m)}(\xi)}.$$
(22)

To be precise, we employed Hermite interpolating polynomial for the computation of  $\alpha_n$  as follows:

**FORM 1.** We computed  $\alpha_n$  as follows:

$$\alpha_n = -\frac{\mathbb{H}_3^{(m+1)}(s_n)}{(m+1)\Omega^{(m)}(s_n)},\tag{23}$$

where

$$\mathbb{H}_{3}(s) = \Omega(s_{n}) + \Omega[s_{n}, s_{n}](s - s_{n}) + \Omega[s_{n}, s_{n}, s_{n}](s - s_{n})^{2} + \dots + \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}] \times$$

$$(s - s_{n})^{m+1} + \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}, y_{n-1}](s - s_{n})^{m+1}(s - y_{n-1})$$

$$+ \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}, y_{n-1}](s - s_{n})^{m+1}(s - y_{n-1})^{2}$$

and

$$\mathbb{H}_{3}^{(m+1)}(s_{n}) = \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}](m+1)! + \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}, y_{n-1}](m+1)! \times (s_{n} - y_{n-1}) + \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}, y_{n-1}, s_{n-1}](m+1)!(s_{n} - y_{n-1})^{2}.$$

**Remark 1.** The Hermite interpolating polynomial  $\mathbb{H}_k(s)$ , k = 3, 4 satisfies the condition  $\mathbb{H}'_k(s_n) = \Omega'(s_n)$ , k = 3, 4.

Now, substituting  $\alpha_n$  from (23) in Equation (7), we obtain the following parametric family of with memory methods, which we denote as PFWM1:

$$y_{n} = s_{n} - m \frac{\Omega(s_{n})}{\Omega'(s_{n}) + \alpha_{n}\Omega(s_{n})}, \ \alpha_{n} = -\frac{\mathbb{H}_{3}^{(m+1)}(s_{n})}{(m+1)\Omega^{(m)}(s_{n})},$$
$$s_{n+1} = y_{n} - mW(t_{n})\frac{\Omega(y_{n})}{\Omega'(y_{n}) + \alpha_{n}\Omega(y_{n})}.$$
(24)

**FORM 2.** We computed  $\alpha_n$  as follows:

$$\alpha_n = -\frac{\mathbb{H}_4^{(m+1)}(s_n)}{(m+1)\Omega^{(m)}(s_n)},\tag{25}$$

where

$$\mathbb{H}_{4}(s) = \Omega(s_{n}) + \Omega[s_{n}, s_{n}](s - s_{n}) + \Omega[s_{n}, s_{n}, s_{n}](s - s_{n})^{2} + \dots + \\ \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}](s - s_{n})^{m+1} + \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}, y_{n-1}](s - s_{n})^{m+1} \times \\ (s - y_{n-1}) + \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}, s_{n-1}](s - s_{n})^{m+1}(s - y_{n-1})^{2} \\ + \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}, s_{n-1}, s_{n-1}](s - s_{n})^{m+1}(s - y_{n-1})^{2}(s - s_{n-1})$$

and

$$\mathbb{H}_{4}^{(m+1)}(s_{n}) = \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}](m+1)! + \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}, y_{n-1}](m+1)! \times \\ (s_{n} - y_{n-1}) + \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}, y_{n-1}, s_{n-1}](m+1)! (s_{n} - y_{n-1})^{2} \\ + \Omega[\underbrace{s_{n}, s_{n}, s_{n}, \dots, s_{n}}_{m+1}, y_{n-1}, y_{n-1}, s_{n-1}](m+1)! (s_{n} - y_{n-1})^{2} (s_{n} - s_{n-1}).$$

Now, substituting  $\alpha_n$  from (25) in Equation (7), we obtain the following parametric family of with memory methods, which we denote as PFWM2:

$$y_{n} = s_{n} - m \frac{\Omega(s_{n})}{\Omega'(s_{n}) + \alpha_{n}\Omega(s_{n})}, \ \alpha_{n} = -\frac{\mathbb{H}_{4}^{(m+1)}(s_{n})}{(m+1)\Omega^{(m)}(s_{n})},$$
$$s_{n+1} = y_{n} - mW(t_{n})\frac{\Omega(y_{n})}{\Omega'(y_{n}) + \alpha_{n}\Omega(y_{n})}.$$
(26)

Thus, PFWM1 and PFWM2 preserve the efficiency properties of the corresponding without memory methods (PFM) and result in the increase of the convergence order of the with memory methods from 5 to at least 7, resulting in at least a 2-order improvement.

Lemma 1. If 
$$\alpha_n = -\frac{\mathbb{H}_3^{(m+1)}(s_n)}{(m+1)\Omega^{(m)}(s_n)}, n = 1, 2, 3, \dots$$
, then the given estimation:  
 $\alpha_n + c_1 \sim c_4 e_{n-1,y}^2 e_{n-1}$ 
(27)

holds, where  $e_n = s_n - \xi$ ,  $e_{n,y} = y_n - \xi$ , and  $c_4$  is an asymptotic constant.

Proof of Lemma 1. The error of Hermite interpolation can be expressed as follows:

$$\Omega(s) - \mathbb{H}_3(s) = \frac{\Omega^{(m+4)}(\delta)}{(m+4)!} (s - s_n)^{(m+1)} (s - y_{n-1})^2 (s - s_{n-1}).$$
(28)

Now, after differentiating (28) (m + 1) times at the point  $s = s_n$  and rearranging, we obtain the expression:

$$\mathbb{H}_{3}^{(m+1)}(s_{n}) = \Omega^{(m+1)}(s_{n}) - \frac{\Omega^{(m+4)}(\delta)}{(m+4)!}(m+1)!(s_{n}-y_{n-1})^{2}(s_{n}-s_{n-1}).$$
(29)

Now, for the multiple roots  $\xi$ , we have the following Taylor series of function  $\Omega$  at the point  $s_n$ :

$$\Omega(s_n) = \frac{\Omega^{(m)}(\xi)}{m!} \Big( e_n^m + c_1 e_n^{m+1} + c_2 e_n^{m+2} + c_3 e_n^{m+3} + O(e_n^{m+4}) \Big).$$
(30)

Then, after differentiating (m + 1) times, we have

$$\Omega^{(m+1)}(s_n) = \frac{\Omega^{(m)}(\xi)}{m!} \Big( (m+1)!c_1 + (m+2)!c_2e_n + O(e_n^2) \Big).$$
(31)

Similarly,

$$\Omega^{(m+4)}(\delta) = \frac{\Omega^{(m)}(\xi)}{m!} \Big( (m+4)!c_4 + (m+5)!c_5e_\delta + O(e_\delta^2) \Big), \tag{32}$$

where  $e_{\delta} = \delta - \xi$ . Thus, using Equations (31) and (32) in (29) and excluding the terms of  $e_n$  and  $e_{\delta}$ , we can obtain

$$\mathbb{H}_{3}^{(m+1)}(s_{n}) \sim \frac{\Omega^{(m)}(\xi)}{m!}(m+1)!c_{1} - \frac{\Omega^{(m)}(\xi)}{m!}c_{4}(m+1)!(s_{n}-y_{n-1})^{2}(s_{n}-s_{n-1}).$$
(33)

After solving this, we have

$$\mathbb{H}_{3}^{(m+1)}(s_{n}) \sim \Omega^{(m)}(\xi) (m+1) \Big( c_{1} - c_{4}(s_{n} - y_{n-1})^{2} (s_{n} - s_{n-1}) \Big), \tag{34}$$

which implies

$$-\frac{\mathbb{H}_{3}^{(m+1)}(s_{n})}{(m+1)\Omega^{(m)}(s_{n})} \sim -\left(c_{1}-c_{4}(s_{n}-y_{n-1})^{2}(s_{n}-s_{n-1})\right)$$
(35)

or

$$\alpha_n \sim -c_1 + c_4 (s_n - y_{n-1})^2 (s_n - s_{n-1}).$$
 (36)

Finally, we have

$$\alpha_n + c_1 \sim c_4 e_{n-1,y}^2 e_{n-1}.$$
(37)

This completes the proof for Lemma 1.  $\Box$ 

**Theorem 2.** Suppose  $\Omega: \mathbb{C} \to \mathbb{C}$  is a function in the neighbourhood of the multiple roots  $\xi$  of  $\Omega(s) = 0$  of multiplicity m > 1. If  $s_0$  is sufficiently close to  $\xi$ , then the R-order of convergence of the iterative method (24) with the parameter  $\alpha_n$  calculated by (23) is at least seven.

**Proof of Theorem 2.** Let the iterative method (IM) generates the sequence of  $\{s_n\}$ , which converges to the root  $\xi$  of  $\Omega(s)$ ; by means of R-order  $O_R(IM, \xi) \ge r$ , we express

$$e_{n+1} \sim D_{n,r} e_n^r \tag{38}$$

and

$$e_n \sim D_{n-1,r} e_{n-1}^r.$$
 (39)

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

$$e_{n+1} \sim D_{n,r} (D_{n-1,r} e_{n-1}^r)^r = D_{n,r} D_{n-1,r}^r e_{n-1}^{r^2}.$$
 (40)

The resulting error expression of the with memory scheme (24) can be obtained using (12) and (9) and the varying parameter  $\alpha_n$  as follows:

$$e_{n,y} = y_n - \xi \sim \frac{\alpha_n + c_1}{m} e_n^2 \tag{41}$$

and

$$e_{n+1} = s_{n+1} - \xi \sim \frac{1}{6(m-1)m^4} (\alpha_n + c_1)^2 \Big( (1-m)(12mc_2 + \alpha_n^2(-6 + W^{(3)}(0))) + 2\alpha_n c_1(m(12 - W^{(3)}(0)) + W^{(3)}(0)) + c_1^2(6 + 6m^2 - m(-12 + W^{(3)}(0) + W^{(3)}(0))) \Big) e_n^5.$$
(42)

Here, the higher-order terms in Equations (41) and (42) are excluded. Now, let *p* be the R-order of convergence of the iterative sequence  $\{y_n\}$ . Then,

$$e_{n,y} \sim D_{n,p} e_n^p \sim D_{n,p} (D_{n-1,r} e_{n-1}^r)^p \sim D_{n,p} D_{n-1,r}^p e_{n-1}^{rp}$$
(43)

and

$$e_{n-1,y} \sim D_{n-1,p} e_{n-1}^p.$$
 (44)

Now, by Equations (37), (41) and (44), we obtain

$$e_{n,y} \sim \frac{\alpha_n + c_1}{m} e_n^2 \sim \left(\frac{c_4 e_{n-1,y}^2 e_{n-1}}{m}\right) (D_{n-1,r} e_{n-1}^r)^2 \\ \sim \left(\frac{c_4 (D_{n-1,p} e_{n-1}^p)^2 e_{n-1}}{m}\right) (D_{n-1,r} e_{n-1}^r)^2 \\ \sim \frac{c_4 D_{n-1,p}^2 D_{n-1,r}^2}{m} e_{n-1}^{2p+2r+1}.$$
(45)

Again, by (37), (42) and (44), we have

$$e_{n+1} \sim \frac{1}{6(m-1)m^4} (\alpha_n + c_1)^2 G_n e_n^5$$
  

$$\sim \frac{1}{6(m-1)m^4} (c_4 e_{n-1,y}^2 e_{n-1})^2 G_n (D_{n-1,r} e_{n-1}^r)^5$$
  

$$\sim \frac{1}{6(m-1)m^4} \left( c_4 (D_{n-1,p} e_{n-1}^p)^2 e_{n-1} \right)^2 G_n (D_{n-1,r} e_{n-1}^r)^5$$
  

$$\sim \frac{1}{6(m-1)m^4} \left( c_4^2 D_{n-1,p}^4 G_n D_{n-1,r}^5 \right) e_{n-1}^{4p+5r+2}, \tag{46}$$

where  $G_n$  originates from (42).

Since r > p, by equating the exponents of  $e_{n-1}$  present in the set of relations (43)–(45) and (40)–(46), we attain the resulting system of equations:

$$rp = 2r + 2p + 1,$$
  
 $r^2 = 5r + 4p + 2.$  (47)

The positive solutions of the system of Equation (47) are r = 7 and p = 3. As a result, the R-order of convergence of the with memory iterative method (24) is at least r = 7.  $\Box$ 

Lemma 2. If 
$$\alpha_n = -\frac{\mathbb{H}_4^{(m+1)}(s_n)}{(m+1)\Omega^{(m)}(s_n)}$$
,  $n = 1, 2, 3, ...,$  then the given estimation:  
 $\alpha_n + c_1 \sim c_5 e_{n-1,y}^2 e_{n-1}^2$ 
(48)

holds, where  $e_n = s_n - \xi$ ,  $e_{n,y} = y_n - \xi$ , and  $c_5$  is an asymptotic constant.

Proof of Lemma 2. The error of Hermite interpolation can be expressed as follows:

$$\Omega(s) - \mathbb{H}_4(s) = \frac{\Omega^{(m+5)}(\delta)}{(m+5)!} (s - s_n)^{(m+1)} (s - y_{n-1})^2 (s - s_{n-1})^2.$$
(49)

Then, after differentiating (49) (m + 1) times at the point  $s = s_n$  and rearranging, we obtain the expression below:

$$\mathbb{H}_{4}^{(m+1)}(s_{n}) = \Omega^{(m+1)}(s_{n}) - \frac{\Omega^{(m+5)}(\delta)}{(m+5)!}(m+1)!(s_{n}-y_{n-1})^{2}(s_{n}-s_{n-1})^{2}.$$
 (50)

Now, for the multiple roots  $\xi$ , we have the following Taylor series of function  $\Omega$  at the point  $s_n$ :

$$\Omega(s_n) = \frac{\Omega^{(m)}(\xi)}{m!} \Big( e_n^m + c_1 e_n^{m+1} + c_2 e_n^{m+2} + c_3 e_n^{m+3} + O(e_n^{m+4}) \Big).$$
(51)

Then, after differentiating (m + 1) times, we have

$$\Omega^{(m+1)}(s_n) = \frac{\Omega^{(m)}(\xi)}{m!} \Big( (m+1)!c_1 + (m+2)!c_2e_n + O(e_n^2) \Big).$$
(52)

Similarly,

$$\Omega^{(m+5)}(\delta) = \frac{\Omega^{(m)}(\xi)}{m!} \Big( (m+5)!c_5 + (m+6)!c_6e_\delta + O(e_\delta^2) \Big),$$
(53)

where  $e_{\delta} = \delta - \xi$ . Thus, using Equations (52) and (53) in (50) and excluding the terms of  $e_n$  and  $e_{\delta}$ , we can obtain

$$\mathbb{H}_{4}^{(m+1)}(s_{n}) \sim \frac{\Omega^{(m)}(\xi)}{m!} (m+1)! c_{1} - \frac{\Omega^{(m)}(\xi)}{m!} c_{5}(m+1)! (s_{n} - y_{n-1})^{2} (s_{n} - s_{n-1})^{2}.$$
(54)

After solving this, we have

$$\mathbb{H}_{4}^{(m+1)}(s_{n}) \sim \Omega^{(m)}(\xi) (m+1) \left( c_{1} - c_{5}(s_{n} - y_{n-1})^{2} (s_{n} - s_{n-1})^{2} \right), \tag{55}$$

which implies

$$-\frac{\mathbb{H}_{4}^{(m+1)}(s_{n})}{(m+1)\Omega^{(m)}(s_{n})} \sim -\left(c_{1}-c_{5}(s_{n}-y_{n-1})^{2}(s_{n}-s_{n-1})^{2}\right)$$
(56)

or

$$\alpha_n \sim -c_1 + c_5(s_n - y_{n-1})^2(s_n - s_{n-1})^2.$$
 (57)

Finally, we have

$$\alpha_n + c_1 \sim c_5 e_{n-1,y}^2 e_{n-1}^2. \tag{58}$$

This completes the proof for Lemma 2.  $\Box$ 

**Theorem 3.** Suppose  $\Omega : \mathbb{C} \to \mathbb{C}$  is a function in the neighbourhood of the multiple roots  $\xi$  of  $\Omega(s) = 0$  of multiplicity m > 1. If  $s_0$  is sufficiently close to  $\xi$ , then the R-order of convergence of the iterative method (26) with the parameter  $\alpha_n$  calculated by (25) is at least  $(7 + \sqrt{57})/2 \approx 7.2749$ .

**Proof of Theorem 3.** Let the iterative method (IM) generates the sequence of  $\{s_n\}$ , which converges to the root  $\xi$  of  $\Omega(s)$ ; by means of R-order  $O_R(IM, \xi) \ge r$ , we express

$$e_{n+1} \sim D_{n,r} e_n^r \tag{59}$$

and

$$e_n \sim D_{n-1,r} e_{n-1}^r. \tag{60}$$

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

$$e_{n+1} \sim D_{n,r} (D_{n-1,r} e_{n-1}^r)^r = D_{n,r} D_{n-1,r}^r e_{n-1}^{r^2}.$$
 (61)

The resulting error expression of the with memory scheme (26) can be obtained using (12) and (9) and the varying parameter  $\alpha_n$  as follows:

$$e_{n,y} = y_n - \xi \sim \frac{\alpha_n + c_1}{m} e_n^2 \tag{62}$$

and

$$e_{n+1} = s_{n+1} - \xi \sim \frac{1}{6(m-1)m^4} (\alpha_n + c_1)^2 \Big( (1-m)(12mc_2 + \alpha_n^2(-6+W^{(3)}(0))) + 2\alpha_n c_1(m(12-W^{(3)}(0)) + W^{(3)}(0)) + c_1^2(6+6m^2 - m(-12+W^{(3)}(0) + W^{(3)}(0))) \Big) e_n^5.$$
(63)

Here, the higher-order terms in Equations (62) and (63) are excluded. Now, let *p* be the R-order convergence of the iterative sequences  $\{y_n\}$ . Then,

$$e_{n,y} \sim D_{n,p} e_n^p \sim D_{n,p} (D_{n-1,r} e_{n-1}^r)^p \sim D_{n,p} D_{n-1,r}^p e_{n-1}^{rp}$$
(64)

and

$$e_{n-1,y} \sim D_{n-1,p} e_{n-1}^p.$$
 (65)

Then, by Equations (58), (62) and (65), we obtain

$$e_{n,y} \sim \frac{\alpha_n + c_1}{m} e_n^2 \sim \left(\frac{c_5 e_{n-1,y}^2 e_{n-1}^2}{m}\right) (D_{n-1,r} e_{n-1}^r)^2 \\ \sim \left(\frac{c_5 (D_{n-1,p} e_{n-1}^p)^2 e_{n-1}^2}{m}\right) (D_{n-1,r} e_{n-1}^r)^2 \\ \sim \frac{c_5 D_{n-1,p}^2 D_{n-1,r}^2}{m} e_{n-1}^{2p+2r+2}.$$
(66)

Again, by (58), (63) and (65), we have

$$e_{n+1} \sim \frac{1}{6(m-1)m^4} (\alpha_n + c_1)^2 G_n e_n^5$$
  

$$\sim \frac{1}{6(m-1)m^4} (c_5 e_{n-1,y}^2 e_{n-1}^2)^2 G_n (D_{n-1,r} e_{n-1}^r)^5$$
  

$$\sim \frac{1}{6(m-1)m^4} \left( c_5 (D_{n-1,p} e_{n-1}^p)^2 e_{n-1}^2 \right)^2 G_n (D_{n-1,r} e_{n-1}^r)^5$$
  

$$\sim \frac{1}{6(m-1)m^4} \left( c_5^2 D_{n-1,p}^4 G_n D_{n-1,r}^5 \right) e_{n-1}^{4p+5r+4}, \tag{67}$$

where  $G_n$  originates from (63).

Since r > p, by equating the exponents of  $e_{n-1}$  present in the set of relations (64)–(66) and (61)–(67), we attain the resulting system of equations:

$$rp = 2r + 2p + 2,$$
  
 $r^2 = 5r + 4p + 4.$  (68)

The positive solutions of the system of Equation (68) are  $r = (7 + \sqrt{57})/2$  and  $p = (5 + \sqrt{57})/4$ . As a result, the R-order of convergence of the with memory iterative method (26) is at least  $r = (7 + \sqrt{57})/2 \approx 7.2749$ .  $\Box$ 

#### 3. Numerical Results

In this section, we assess the performance and computational efficiency of the newly developed families of methods PFM, PFWM1 and PFWM2, which were discussed in Section 2. As for the weight function  $W(t_n)$ , we use the particular case given in Equation (18) throughout the whole computation. Additionally, we compare these methods with other similar approaches described in existing literature. The objective of this analysis is to provide a comprehensive evaluation of the proposed families of methods and to validate their theoretical results. By conducting this comparison, we aim to gain a better understanding of the effectiveness and practicality of these new methods and their potential utility for various applications.

Now, let us consider some existing methods for finding multiple roots available in literature for the comparison.

A subcase of the third-order method given by Vinay Kanwar et al. [7], which was developed in March, 2023. The method is denoted as VKM, and the expression is given below:

$$s_{n+1} = s_n - \left(\frac{2}{2 - L_f}\right) \frac{m\Omega(s_n)}{\Omega'(s_n) - m\lambda\Omega(s_n)},\tag{69}$$

where 
$$\lambda = 1$$
,  $L_f = \frac{m\Omega(s_n) \left( \Omega''(s_n) + m\lambda^2 \Omega(s_n) \right) - \Omega'(s_n)^2 (m-1) - 2m\lambda\Omega(s_n)\Omega'(s_n)}{\left( \Omega'(s_n) - m\lambda\Omega(s_n) \right)^2}$ .

A particular case of the with memory methods from Equation (3), which was developed in February 2023 [11]. The method is denoted as XZM, and it is given below:

$$\alpha_{n} = -\frac{1}{m+1} \frac{N_{2}^{(m+1)}(s_{n})}{\Omega^{(m)}(s_{n})},$$

$$y_{n} = s_{n} - m \frac{\Omega(s_{n})}{\Omega'(s_{n}) + \alpha_{n}\Omega(s_{n})},$$

$$s_{n+1} = y_{n} - mu_{n} \left(\frac{1+4u_{n}}{(1+u_{n})^{2}}\right) \frac{\Omega(s_{n})}{\Omega'(s_{n}) + 2\alpha_{n}\Omega(s_{n})},$$
(70)

where  $u_n = \left(\frac{\Omega(y_n)}{\Omega(s_n)}\right)^{\frac{1}{m}}$ .

A subcase of the fifth order methods given by Chanu-Panday-Dwivedi [14], which was developed in the year 2021. The method is denoted as CPDM, and it is given below:

$$y_{n} = s_{n} + m \frac{\Omega(s_{n})}{\Omega'(s_{n})},$$

$$z_{n} = s_{n} - \frac{m}{2^{m}} \frac{\Omega(y_{n})}{\Omega'(s_{n})},$$

$$s_{n+1} = z_{n} - m \left(1 - \frac{2}{m} + \frac{4}{m^{2}} + \left(\frac{4}{m} - \frac{8}{m^{2}}\right)h + \left(\frac{4}{m^{2}} - \frac{2}{m}\right)h^{2}\right) \frac{\Omega(z_{n})}{\Omega'(z_{n})},$$
(71)

where  $h = 2^m \frac{\Omega(s_n)}{\Omega(y_n)}$ .

A subcase of the fifth order methods given by Sharma-Arora [10], which was developed in the year 2021. The method is denoted as SAM, and it is given below:

$$y_{n} = s_{n} - m \frac{\Omega(s_{n})}{\Omega'(s_{n})},$$

$$s_{n+1} = y_{n} - m \left(\frac{1 + \left(\frac{\Omega(y_{n})}{\Omega(s_{n})}\right)^{\frac{1}{m}} + \left(\frac{\Omega(y_{n})}{\Omega(s_{n})}\right)^{\frac{2}{m}}}{1 + \left(\frac{\Omega(y_{n})}{\Omega(s_{n})}\right)^{\frac{1}{m}}}\right) \frac{\Omega(y_{n})}{\Omega'(y_{n})}.$$
(72)

A subcase of the fifth-order methods given by Singh-Arora-Jäntschi [15], which was developed in January 2023. The method is denoted as SAJM, and it is given below:

$$y_{n} = s_{n} - m \frac{\Omega(s_{n})}{\Omega'(s_{n})},$$
  

$$s_{n+1} = y_{n} - m(1 + v^{2}) \frac{\Omega(y_{n})}{\Omega'(y_{n})},$$
(73)

where  $v = \frac{u}{1+\lambda u}$ ,  $u = \left(\frac{\Omega(y_n)}{\Omega(s_n)}\right)^{\frac{1}{m}}$ ,  $\lambda = -1$ .

A subcase of the sixth-order methods given by Geum-Kim-Neta [8], which was developed in 2015. The method is denoted as GKNM, and it is given below:

$$y_n = s_n - m \frac{\Omega(s_n)}{\Omega'(s_n)},$$
  

$$s_{n+1} = y_n - \left(\frac{m + a_1 u}{(1 + b_1 u + b_2 u^2)^2}\right) \left(\frac{1}{1 + 2(m-1)t}\right) \frac{\Omega(y_n)}{\Omega'(y_n)},$$
(74)

where  $u = \left(\frac{\Omega(y_n)}{\Omega(s_n)}\right)^{\frac{1}{m}}$ ,  $t = \left(\frac{\Omega'(y_n)}{\Omega'(s_n)}\right)^{\frac{1}{m-1}}$ ,  $a_1 = \frac{2m(4m^4 - 16m^3 + 31m^2 - 30m + 13)}{(m-1)(4m^2 - 8m + 7)}$ ,  $b_1 = \frac{4(2m^2 - 4m + 3)}{(m-1)(4m^2 - 8m + 7)}$ ,  $b_2 = -\frac{4m^2 - 8m + 3}{4m^2 - 8m + 7}$ .

We performed all numerical tests using Mathematica 12.2, a multi-precision arithmetic programming software. To ensure consistency across all test functions, we set the initial parameter value to  $\alpha_0 = -1$  to initiate the iteration.

The test functions consisted of some academic and real-life engineering examples, each with multiple roots ( $\xi$ ) and corresponding initial guesses ( $s_0$ ). These functions and their associated roots and initial guesses are listed below.

**Example 1.** A standard academic test function given by

$$\Omega_1(s) = (4 + 3\sin s - 2s^2)^4.$$

It has the multiple roots  $\xi = 2$  with multiplicity m = 4. We used  $s_0 = 2.2$  as the initial guess, and the results are displayed in Table 1.

Methods	п	$ s_1 - s_0 $	$ s_2 - s_1 $	$ s_3 - s_2 $	$ \Omega(s_3) $	COC
VKM	6	0.33008	$1.5207  imes 10^{-2}$	$9.7604\times10^{-7}$	$1.9584  imes 10^{-71}$	3.0000
XZM	6	0.34505	$2.3864 imes10^{-4}$	$2.6415  imes 10^{-10}$	$1.4426  imes 10^{-194}$	1.9972
CPDM	5	0.34522	$6.5995  imes 10^{-5}$	$1.5798 \times 10^{-22}$	$1.1084  imes 10^{-436}$	5.0000
SAM	5	0.34504	$2.5032 imes10^{-4}$	$1.5353  imes 10^{-19}$	$1.4727  imes 10^{-376}$	5.0000
SAJM	5	0.34514	$1.4499\times 10^{-4}$	$4.2323  imes 10^{-21}$	$3.0144  imes 10^{-409}$	5.0000
GKNM	5	0.34517	$1.2267 imes10^{-4}$	$1.7184  imes 10^{-24}$	$1.3290  imes 10^{-568}$	6.0000
PFM	5	0.34528	$1.3324 imes10^{-5}$	$5.4270  imes 10^{-27}$	$6.3786  imes 10^{-530}$	5.0000
PFWM1	4	0.34528	$1.3324 imes10^{-5}$	$5.3785  imes 10^{-35}$	$4.0310  imes 10^{-957}$	7.0000
PFWM2	4	0.34528	$1.3324\times 10^{-5}$	$4.9818\times10^{-35}$	$3.2264  imes 10^{-958}$	7.0000

**Table 1.** Comparison results of the methods for  $\Omega_1(s)$ .

**Example 2.** A standard academic test function given by

$$\Omega_2(s) = \left(\sin^2 s + s\right)^5.$$

It has the multiple roots  $\xi = 0$  with multiplicity m = 5. We used  $s_0 = 0.6$  as the initial guess, and the results are displayed in Table 2.

**Table 2.** Comparison results of the methods for  $\Omega_2(s)$ .

Methods	n	$ s_1 - s_0 $	$ s_2 - s_1 $	$ s_3 - s_2 $	$ \Omega(s_3) $	COC
VKM	7	0.45485	$1.4360\times10^{-1}$	$1.5518\times 10^{-3}$	$2.2780  imes 10^{-44}$	3.0000
XZM	6	0.58388	$1.6122  imes 10^{-2}$	$2.3381  imes 10^{-9}$	$3.8284  imes 10^{-107}$	3.0017
CPDM	5	0.59039	$9.6112  imes 10^{-3}$	$8.7442  imes 10^{-10}$	$8.6888  imes 10^{-222}$	5.0000
SAM	5	0.59002	$9.9759  imes 10^{-3}$	$4.5553  imes 10^{-10}$	$9.0715  imes 10^{-231}$	5.0000
SAJM	5	0.59138	$8.6160  imes 10^{-3}$	$9.0956  imes 10^{-11}$	$2.9921  imes 10^{-250}$	5.0000
GKNM	6	0.59236	$7.6391\times10^{-3}$	$7.8345  imes 10^{-12}$	$1.2782  imes 10^{-325}$	6.0000

Methods	n	$ s_1 - s_0 $	$ s_2 - s_1 $	$ s_3 - s_2 $	$ \Omega(s_3) $	COC
PFM PFWM1 PFWM2	5 5 5	0.59660 0.59660 0.59660	$\begin{array}{c} 3.4028 \times 10^{-3} \\ 3.4028 \times 10^{-3} \\ 3.4028 \times 10^{-3} \end{array}$	$\begin{array}{c} 1.1488 \times 10^{-12} \\ 1.2404 \times 10^{-15} \\ 1.0346 \times 10^{-15} \end{array}$	$\begin{array}{l} 3.7027 \times 10^{-297} \\ 1.3157 \times 10^{-510} \\ 1.8921 \times 10^{-513} \end{array}$	5.0000 7.0000 7.0000

Table 2. Cont.

**Example 3.** A standard academic test function given by

$$\Omega_3(s) = \left(e^{-s^2 + s + 3} - s + 2\right)^9.$$

It has the multiple roots  $\xi \approx 2.4905398276083051$  with multiplicity m = 9. We used  $s_0 = 2.6$  as the initial guess, and the results are displayed in Table 3.

Methods	n	$ s_1 - s_0 $	$ s_2 - s_1 $	$ s_3 - s_2 $	$ \Omega(s_3) $	COC
VKM	6	0.10680	$2.6650 \times 10^{-3}$	$3.5814 imes10^{-8}$	$4.7372  imes 10^{-195}$	3.0000
XZM	7	0.14985	$4.1261  imes 10^{-2}$	$7.2813 imes10^{-6}$	$1.8238  imes 10^{-107}$	2.9992
CPDM	5	0.10774	$1.7225  imes 10^{-3}$	$2.0111  imes 10^{-12}$	$9.8289  imes 10^{-504}$	5.0000
SAM	5	0.10948	$1.6599 imes10^{-4}$	$2.2757  imes 10^{-15}$	$1.6903  imes 10^{-684}$	3.9991
SAJM	5	0.10941	$3.0955 imes10^{-4}$	$2.4209  imes 10^{-14}$	$4.9884  imes 10^{-611}$	3.9991
GKNM	6	0.11076	$3.5248 imes10^{-3}$	$1.5277  imes 10^{-7}$	$3.9790  imes 10^{-178}$	7.4893
PFM	5	0.10931	$1.5165 imes10^{-4}$	$5.5816  imes 10^{-19}$	$2.6128  imes 10^{-810}$	5.0000
PFWM1	4	0.10931	$1.5165 imes10^{-4}$	$2.0972 \times 10^{-24}$	$1.6827  imes 10^{-1450}$	6.9999
PFWM2	4	0.10931	$1.5165\times 10^{-4}$	$3.4644  imes 10^{-24}$	$1.3666  imes 10^{-1444}$	7.0000

**Table 3.** Comparison results of the methods for  $\Omega_3(s)$ .

**Example 4** (Manning equation for fluid dynamics [16]). *The equation is given below:* 

$$\Omega_4(s) = \left[\tan^{-1}\left(\frac{\sqrt{5}}{2}\right) - \tan^{-1}\left(\sqrt{s^2 - 1}\right) + \sqrt{6}\left(\tan^{-1}\left(\sqrt{\frac{s^2 - 1}{6}}\right) - \tan^{-1}\left(\frac{1}{2}\sqrt{\frac{5}{6}}\right)\right) - \frac{11}{63}\right]^7.$$
(75)

It has the multiple roots  $\xi \approx 1.8411294068501996$  with multiplicity m = 7. We used  $s_0 = 1.2$  as the initial guess, and the results are displayed in Table 4.

**Table 4.** Comparison results of the methods for  $\Omega_4(s)$ .

	Methods	п	$ s_1 - s_0 $	$ s_2 - s_1 $	$ s_3 - s_2 $	$ \Omega(s_3) $	COC
_	VKM	7	0.50271	$1.3682 imes10^{-1}$	$1.5989  imes 10^{-3}$	$1.1713  imes 10^{-62}$	3.0000
	XZM	6	0.70523	$6.5936  imes 10^{-2}$	$2.8688 imes10^{-4}$	$1.0602  imes 10^{-75}$	2.0024
	CPDM	5	0.64191	$1.0894 imes10^{-3}$	$3.6165  imes 10^{-17}$	$1.0565  imes 10^{-589}$	5.0000
	SAM	5	0.63813	$3.5058 imes10^{-3}$	$2.7647  imes 10^{-13}$	$2.2144  imes 10^{-375}$	3.9978
	SAJM	5	0.63791	$4.6433  imes 10^{-3}$	$8.5166  imes 10^{-13}$	$1.0630  imes 10^{-361}$	3.9978
	GKNM	6	0.44125	$5.4541 imes10^{-4}$	$6.4964  imes 10^{-12}$	$1.5113  imes 10^{-247}$	3.0028
	PFM	5	0.64075	$3.7793 imes10^{-4}$	$1.4359  imes 10^{-19}$	$1.9352  imes 10^{-674}$	5.0000
	PFWM1	4	0.64075	$3.7793 imes10^{-4}$	$2.0608  imes 10^{-25}$	$1.1797  imes 10^{-1215}$	7.0000
	PFWM2	4	0.64075	$3.7793 imes10^{-4}$	$1.2603  imes 10^{-25}$	$3.5869  imes 10^{-1229}$	7.0000

**Example 5** (Van der Waals equation of state [17]). *The Van der Waals equation of state is a modification of the ideal gas law, which takes into account the forces of attraction between gas molecules and the finite size of the molecules themselves. It is given by the following equation:* 

$$\left(p + \frac{an^2}{v^2}\right)(v - nb) = nRT,\tag{76}$$

where p is the pressure, v is the volume, R is the gas constant, T is the temperature, n is the number of moles, a is a parameter that represents the strength of the intermolecular forces, and b

*is a parameter that represents the size of the molecules. This equation is useful for describing the behaviour of real gases, which deviate from the ideal gas law at high pressures and low temperatures.* 

*To solve for the volume v in terms of the other parameters, we can rearrange the Van der Waals equation of state to obtain a cubic equation:* 

$$pv^{3} - (nbp + nRT)v^{2} + an^{2}v = abn^{3}.$$
(77)

Let us consider a gas with n = 0.1807 moles, a = 278.3 atm  $L^2/mol^2$ , and b = 3.2104 L/mol, at a pressure of 1 atm and a temperature of 313 K. Using the universal gas constant R = 0.08206 atm L/mol K, we can plug these values into the cubic equation above and solve for the volume v. Then, we have

$$\Omega_5(s) = s^3 - 5.22s^2 + 9.0825s - 5.2675, \tag{78}$$

where s = v, yielding the multiple roots  $\xi = 1.75$  with multiplicity m = 2. We used  $s_0 = 2.5$  as the initial guess, and the results are displayed in Table 5.

Methods	n	$ s_1-s_0 $	$ s_2 - s_1 $	$ s_3 - s_2 $	$ \Omega(s_3) $	COC
VKM	9	0.54909	$1.7090 imes10^{-1}$	$2.8013\times10^{-2}$	$1.2682  imes 10^{-7}$	3.0000
XZM	6	0.71596	$3.3325 \times 10^{-2}$	$7.1065 imes10^{-4}$	$3.0773  imes 10^{-23}$	5.0000
CPDM	6	0.70133	$4.8193  imes 10^{-2}$	$4.7547 imes10^{-4}$	$3.0603  imes 10^{-28}$	6.0000
SAM	6	0.68022	$6.7340  imes 10^{-2}$	$2.4441 imes10^{-3}$	$2.0299  imes 10^{-17}$	5.0000
SAJM	6	0.68454	$6.3596  imes 10^{-2}$	$1.8660  imes 10^{-3}$	$4.8728  imes 10^{-19}$	5.0000
GKNM	6	0.68637	$6.1983  imes 10^{-2}$	$1.6473 imes10^{-3}$	$1.0720  imes 10^{-20}$	6.0000
PFM	6	0.73275	$1.7174  imes 10^{-2}$	$7.3647\times10^{-5}$	$5.5190  imes 10^{-32}$	5.0000
PFWM1	5	0.73277	$1.7235  imes 10^{-2}$	$1.2844 imes10^{-5}$	$4.6789  imes 10^{-53}$	7.0000
PFWM2	5	0.73275	$1.7235 \times 10^{-2}$	$1.2844\times 10^{-5}$	$4.6789  imes 10^{-53}$	7.0000

**Table 5.** Comparison results of the methods for  $\Omega_5(s)$ .

**Example 6** (Beam designing model [16]). *Here, we deal with a beam that is positioned at an angle towards the edge of a cubic box. The length of the beam is denoted as "r" units, while each side of the box measures one unit. The beam is inclined in a manner such that one end touches the wall, while the other end touches the floor, as depicted in Figure 1.* 



Figure 1. Beam designing model [16].

The objective of this problem is to determine the distance between the base of the wall and the floor along the bottom of the beam. Let us assume that "y" is the distance between the beam and the floor, measured along the edge of the box. Similarly, "s = x" is the distance between the bottom of the box and the beam. For a specific value of "r", the following equation holds true:

$$\Omega_6(s) = s^4 + 4s^3 - 24s^2 + 16s + 16.$$
<sup>(79)</sup>

The equation has a non-negative root, namely  $\xi = 2$ , which has a multiplicity of m = 2. We used  $s_0 = 1.2$  as the initial guess, and the results are displayed in Table 6.

**Table 6.** Comparison results of the methods for  $\Omega_6(s)$ .

Methods	п	$ s_1 - s_0 $	$ s_2 - s_1 $	$ s_3 - s_2 $	$ \Omega(s_3) $	COC
VKM	7	0.71037	$8.9411 imes10^{-2}$	$2.1557  imes 10^{-4}$	$2.5102  imes 10^{-22}$	3.0000
XZM	6	0.79736	$2.6375  imes 10^{-3}$	$1.7127\times 10^{-8}$	$2.7065  imes 10^{-82}$	1.9986
CPDM	5	0.82797	$2.7966  imes 10^{-2}$	$1.8836  imes 10^{-12}$	$8.5982  imes 10^{-145}$	6.0000
SAM	5	0.76907	$3.0928  imes 10^{-2}$	$3.9147\times10^{-10}$	$3.4391  imes 10^{-97}$	5.0000
SAJM	6	0.48555	$3.1425 imes10^{-1}$	$1.9359\times10^{-4}$	$1.0868  imes 10^{-39}$	5.0000
GKNM	6	1.5402	$7.3961  imes 10^{-1}$	$5.6854 imes10^{-4}$	$2.7009  imes 10^{-41}$	6.0000
PFM	5	0.80258	$2.5811  imes 10^{-3}$	$1.6437  imes 10^{-15}$	$7.0888  imes 10^{-151}$	5.0000
PFWM1	5	0.80274	$2.5811  imes 10^{-3}$	$2.7609  imes 10^{-21}$	$4.6947  imes 10^{-292}$	7.0000
PFWM2	5	0.80258	$2.5811  imes 10^{-3}$	$2.7609  imes 10^{-21}$	$4.6947  imes 10^{-292}$	7.0000

Tables 1-6 summarise the numerical results of the compared methods on the test examples. In these tables, we report the number of iterations (*n*) needed to converge to the multiple roots with a stopping criterion of

$$|s_n - s_{n-1}| + |\Omega(s_n)| < 10^{-150}$$
(80)

along with the estimated error in consecutive iterations  $|s_n - s_{n-1}|$  during the first three iterations. Additionally, we provide the absolute residual error of the function  $|\Omega(s_n)|$  and the computational order of convergence (COC), which we calculated using the following formula [18]:

$$COC = \frac{\log |\Omega(s_n)/\Omega(s_{n-1})|}{\log |\Omega(s_{n-1})/\Omega(s_{n-2})|}.$$
(81)

The computational results presented in Tables 1–6 demonstrate that our proposed methods outperformed existing methods in terms of the error in consecutive iterations and the residual error with the minimal number of iterations required for convergence towards the root after the stopping criterion (80) is satisfied. In Figure 2, we provide the comparison of the methods based on the error in the consecutive iterations,  $|s_n - s_{n-1}|$ , after the first three iterations. While the methods XZM, CPDM, SAM, SAJM, and GKNM have better results than the method VKM, they do not always retain their order of convergence. In contrast, our proposed methods PFM, PFWM1, and PFWM2 demonstrated higher precision and maintained their order of convergence, thus confirming their theoretical results. Overall, our proposed methods offer improved performance and accuracy compared to existing methods.



**Figure 2.** Comparison of the methods based on the error in consecutive iterations,  $|s_n - s_{n-1}|$ , after the first three iterations.

## 4. Conclusions

We introduced new parametric families of two-point with and without memory methods for solving nonlinear equations with multiple zeros. These methods are based on the weight function approach and self-accelerating technique. Our proposed family of the without memory methods, PFM, achieved fifth-order convergence with two function and two derivative evaluations per iteration, while the with memory methods PFWM1 and PFWM2 achieved R-orders of convergence of 7 and 7.2749, respectively, using a self-accelerating parameter. Additionally, the use of an accelerating parameter not only improved the convergence order, but also increased the efficiency and accuracy. We verified our theoretical results through numerous numerical examples and compared our methods with existing methods. The results demonstrated the robustness and superior performance of our proposed methods, with fewer iterations required to converge to the root, minimal residual errors, and minimal errors in consecutive iterations.

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

**Funding:** This work was supported by the University Grants Commission (UGC) NET-JRF fellowship UGC-Ref. No.: 1187/(CSIR-UGC NET JUNE 2019).

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

**Acknowledgments:** The authors gratefully acknowledge the University Grants Commission (UGC), New Delhi, India, for providing financial assistance to carry out this work.

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

## References

- Behl, R.; Salimi, M.; Ferrara, M.; Sharifi, S.; Alharbi, S.K. Some Real-Life Applications of a Newly Constructed Derivative Free Iterative Scheme. *Symmetry* 2019, *11*, 239. [CrossRef]
- Chanu, W.H.; Panday, S.; Thangkhenpau, G. Development of Optimal Iterative Methods with Their Applications and Basins of Attraction. Symmetry 2022, 14, 2020. [CrossRef]
- Naseem, A.; Rehman, M.A.; Abdeljawad, T.A. Novel Root-Finding Algorithm With Engineering Applications and its Dynamics via Computer Technology. *IEEE Access* 2022, 10, 19677–19684. [CrossRef]
- 4. Panday, S.; Sharma, A.; Thangkhenpau, G. Optimal fourth and eighth-order iterative methods for non-linear equations. *J. Appl. Math. Comput.* **2023**, *69*, 953–971. [CrossRef]
- Abdul-Hassan, N.Y.; Ali, A.H.; Park, C. A new fifth-order iterative method free from second derivative for solving nonlinear equations. J. Appl. Math. Comput. 2021, 68, 2877–2886. [CrossRef]
- 6. Thangkhenpau, G.; Panday, S. Optimal Eight Order Derivative-Free Family of Iterative Methods for Solving Nonlinear Equations. *IAENG Int. J. Comput. Sci.* 2023, *50*, 335–341.
- Kanwar, V.; Cordero, A.; Torregrosa, J.R.; Rajput, M.; Behl, R. A New Third-Order Family of Multiple Root-Findings Based on Exponential Fitted Curve. *Algorithms* 2023, 16, 156. [CrossRef]
- Geum, Y.H.; Kim, Y.I.; Neta, B. A class of two-point sixth-order multiple-zero finders of modified double-Newton type and their dynamics. *Appl. Math. Comput.* 2015, 270, 387–400. [CrossRef]
- 9. Behl, R. A Derivative Free Fourth-Order Optimal Scheme for Applied Science Problems. Mathematics 2022, 10, 1372. [CrossRef]
- Sharma, J.R.; Arora, H. A Family of Fifth-Order Iterative Methods for Finding Multiple Roots of Nonlinear Equations. *Numer. Anal. Appl.* 2021, 14, 168–199. [CrossRef]
- 11. Zhou, X.; Liu, B. Iterative methods for multiple roots with memory using self-accelerating technique. J. Comput. Appl. Math. 2023, 428, 115181. [CrossRef]
- 12. Kanwar, V.; Bhatia, S.; Kansal, M. New optimal class of higher-order methods for multiple roots, permitting  $f'(x_n) = 0$ . *Appl. Math. Comput.* **2013**, 222, 564–574. [CrossRef]
- 13. Zafar, F.; Cordero, A.; Torregrosa, J.R. Stability analysis of a family of optimal fourth-order methods for multiple roots. *Numer. Algor.* **2019**, *81*, 947–981. [CrossRef]
- 14. Chanu, W.H.; Panday, S.; Dwivedi, M. New Fifth Order Iterative Method for Finding Multiple Root of Nonlinear Function. *Eng. Lett.* **2021**, *29*, 942–947.

- 15. Singh, T.; Arora, H.; Jäntschi, L. A Family of Higher Order Scheme for Multiple Roots. Symmetry 2023, 15, 228. [CrossRef]
- 16. Sharma, J.R.; Kumar, D.; Cattani, C. An Efficient Class of Weighted-Newton Multiple Root Solvers with Seventh Order Convergence. *Symmetry* **2019**, *11*, 1054. [CrossRef]
- 17. Zafar, F.; Cordero, A.; Rizvi, D.E.Z.; Torregrosa, J.R. An optimal eighth order derivative free multiple root finding scheme and its dynamics. *AIMS Math.* 2023, *8*, 8478–8503. [CrossRef]
- Petković, M.S. Remarks on "On a general class of multipoint root-finding methods of high computational efficiency". SIAM J. Numer. Math 2011, 49, 1317–1319. [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.