



An Optimal Derivative-Free Ostrowski's Scheme for Multiple Roots of Nonlinear Equations

Ramandeep Behl¹, Samaher Khalaf Alharbi^{1,2}, Fouad Othman Mallawi¹ and Mehdi Salimi^{3,4,*} ¹

- ¹ Department of Mathematics, King Abdulaziz University, Jeddah 21589, Saudi Arabia; ramanbehl87@yahoo.in (R.B.); samaher271@gmail.com or sk.alharbi@qu.edu.sa (S.K.A.); fmallawi@hotmail.com (F.O.M.)
- ² Department of Mathematics, College of Sciences and Arts in Al-Rass, Qassim University, Al-Rass 51921, Saudi Arabia
- ³ Department of Mathematics & Statistics, McMaster University, Hamilton, ON L8S 4L8, Canada
- ⁴ Center for Dynamics, Faculty of Mathematics, Technische Universität Dresden, 01062 Dresden, Germany
- * Correspondence: mehdi.salimi@mcmaster.ca or msalimi1@yahoo.com

Received: 13 September 2020; Accepted: 10 October 2020; Published: 16 October 2020



Abstract: Finding higher-order optimal derivative-free methods for multiple roots ($m \ge 2$) of nonlinear expressions is one of the most fascinating and difficult problems in the area of numerical analysis and Computational mathematics. In this study, we introduce a new fourth order optimal family of Ostrowski's method without derivatives for multiple roots of nonlinear equations. Initially the convergence analysis is performed for particular values of multiple roots—afterwards it concludes in general form. Moreover, the applicability and comparison demonstrated on three real life problems (e.g., Continuous stirred tank reactor (CSTR), Plank's radiation and Van der Waals equation of state) and two standard academic examples that contain the clustering of roots and higher-order multiplicity (m = 100) problems, with existing methods. Finally, we observe from the computational results that our methods consume the lowest CPU timing as compared to the existing ones. This illustrates the theoretical outcomes to a great extent of this study.

Keywords: nonlinear equation; King–Traub conjecture; multiple root; optimal iterative method; efficiency index

MSC: 65G99; 65H10

1. Introduction

Obtaining the new higher-order derivative free techniques for the multiple solutions (with known multiplicity $m \ge 2$) of f(x) ($f : \mathbb{D} \subset \mathbb{C} \to \mathbb{C}$ a holomorphic map in \mathbb{D} containing the required zero) is one of the robust and difficult issues in the field of numerical analysis and computational mathematics. Finding the derivatives of function/s are either complicated, time consuming or impossible. So, higher-order derivative-free techniques are in high demand and attracting the scholars in this area.

In recent years, some scholars [1–6] suggested many derivative free methods that can handle the simple roots for nonlinear equations. No doubts these methods also work for the multiple roots but are impossible for maintaining the same order of convergence unless some changes/modifications are done in their original substeps. Generally, we obtain the linear order of convergence whenever we used simple root finding techniques for the multiple roots of nonlinear equations.

Mathematics 2020, 8, 1809

Very recently, some researchers proposed the following optimal fourth-order derivative free techniques for the multiple solutions of nonlinear equations, when the multiplicity m is known in advance.

In 2015, Hueso et al. [7] suggested the flowing fourth-order derivative-free technique for multiple roots

$$y_{\lambda} = x_{\lambda} - b \frac{f(x_{\lambda})}{f[\mu_{\lambda}, x_{\lambda}]},$$

$$x_{\lambda+1} = x_{\lambda} - \left(a_1 + a_2 h(y_{\lambda}, x_{\lambda}) + a_3 h(x_{\lambda}, y_{\lambda}) + a_4 h(y_{\lambda}, x_{\lambda})^2\right) \frac{f(x_{\lambda})}{f[\mu_{\lambda}, x_{\lambda}]},$$
(1)

where $\mu_{\lambda} = x_{\lambda} + f(x_{\lambda})^{q}$, $q \in \mathbb{R}$ with $h(x_{\lambda}, y_{\lambda}) = \frac{f[y_{\lambda} + f(y_{\lambda})^{q}, y_{\lambda}]}{f[x_{\lambda} + f(x_{\lambda})^{q}, x_{\lambda}]}$. The scheme (1) attains the optimal fourth-order convergence for $\begin{cases} q = 1, & \text{for each } m \geq 4 \\ q \geq 2, & \text{for all } m \geq 2 \end{cases}$, where *m* is known multiplicity of the required zero.

In 2019, Sharma et al. [8], developed an iteration function [9–16] of optimal fourth-order, which is given by

$$z_{\lambda} = t_{\lambda} - m \frac{f(t_{\lambda})}{f[s_{\lambda}, t_{\lambda}]},$$

$$t_{\lambda+1} = z_{\lambda} - H(x_{\lambda}, y_{\lambda}) \frac{f(t_{\lambda})}{f[s_{\lambda}, t_{\lambda}]},$$
(2)

where $s_{\lambda} = t_{\lambda} + \beta f(t_{\lambda})$, $x_{\lambda} = \left(\frac{f(z_{\lambda})}{f(t_{\lambda})}\right)^{\frac{1}{m}}$, $y_{\lambda} = \left(\frac{f(z_{\lambda})}{f(s_{\lambda})}\right)^{\frac{1}{m}}$ and the conditions on *H* can be found in [8]. In 2020, Sharma et al. [17], proposed the following optimal fourth-order scheme

$$z_{\lambda} = u_{\lambda} - m \frac{f(u_{\lambda})}{f[v_{\lambda}, u_{\lambda}]},$$

$$u_{\lambda+1} = z_{\lambda} - G(h_{\lambda}) \left(\frac{1}{y_{\lambda}} + 1\right) \frac{f(u_{\lambda})}{f[v_{\lambda}, u_{\lambda}]},$$
(3)

where $v_{\lambda} = u_{\lambda} + \beta f(u_{\lambda})$, $x_{\lambda} = \left(\frac{f(z_{\lambda})}{f(u_{\lambda})}\right)^{\frac{1}{m}}$ and $y_{\lambda} = \left(\frac{f(v_{\lambda})}{f(u_{\lambda})}\right)^{\frac{1}{m}}$, with $h_{\lambda} = \frac{x_{\lambda}}{x_{\lambda}+1}$. In addition, the weight function $G : \mathbb{C}^2 \to \mathbb{C}$ is an analytic map in the neighborhood of origin and the conditions on G can be found in [17].

Very recently in 2020, Kumar et al. [18], provided an optimal fourth-order derivative free iteration function, which is defined as follows:

$$w_{\lambda} = u_{\lambda} - m \frac{f(x_{\lambda})}{f[v_{\lambda}, u_{\lambda}]},$$

$$u_{\lambda+1} = w_{\lambda} - \frac{s_{\lambda}}{\eta_1 + \eta_2 s_{\lambda}} \frac{f(u_{\lambda})}{\eta_3 f[v_{\lambda}, u_{\lambda}] + \eta_4 f[w_{\lambda}, v_{\lambda}]},$$
(4)

where $\eta_1, \eta_2, \eta_3, \eta_4$ are disposable parameters with $v_{\lambda} = x_{\lambda} + \beta f(u_{\lambda}), s_{\lambda} = \left(\frac{f(w_{\lambda})}{f(u_{\lambda})}\right)^{\frac{1}{m}}$.

Moving in the same flow, we intend to develop a new fourth order family of Ostrowski's method without derivatives for multiple roots of nonlinear equations for the first time. The proposed scheme consumes only three values of the involved function f and attains the optimal order of convergence in the sense of classical Kung–Traub conjecture [19]. Our methods have not only simple body structure but also consume the lowest CPU time as compared to the existing ones. In addition, we propose a main theorem which illustrates the fourth-order convergence when the multiplicity of roots (m) is known in advance. A numerical exhibition of our scheme is also demonstrated on the real life problems, such as Continuous stirred tank reactor (CSTR), Plank's radiation and Van der Waals equation of state.

Moreover, we also consider the root clustering problem and an example with higher order multiplicity m = 100 in order to demonstrate the behavior of our methods, as compared the earlier methods.

2. Construction of Higher-Order Scheme

Here, we construct an optimal fourth-order family of Ostrowski's method [20,21] for multiple zeros ($m \ge 2$) with simple and compact body structure, which is defined by

$$z_{\lambda} = x_{\lambda} - m \frac{f(x_{\lambda})}{f[\mu_{\lambda}, x_{\lambda}]},$$

$$x_{\lambda+1} = z_{\lambda} + \frac{(z_{\lambda} - x_{\lambda})(\alpha s_{\lambda} + \beta t_{\lambda})}{1 - 2s_{\lambda}},$$
(5)

where $\mu_{\lambda} = x_{\lambda} + \kappa f(x_{\lambda}), \ \kappa \neq 0 \in \mathbb{R}$ is any finite real number and $m \ge 2$ is the known multiplicity of the required zero. In addition, $f[\mu_{\lambda}, x_{\lambda}]$ is finite difference of order one $f[\mu_{\lambda}, x_{\lambda}] = \frac{f(\mu_{\lambda}) - f(x_{\lambda})}{\mu_{\lambda} - x_{\lambda}}$. Moreover, $s_{\lambda} = \left(\frac{f(z_{\lambda})}{f(x_{\lambda})}\right)^{\frac{1}{m}}$ and $t_{\lambda} = \left(\frac{f(z_{\lambda})}{f(\mu_{\lambda})}\right)^{\frac{1}{m}}$ are two multi-valued functions. Suppose their principal analytic branches (see [22]), s_{λ} as a principal root given by $s_{\lambda} = \exp\left[\frac{1}{m}\log\left(\frac{f(z_{\lambda})}{f(x_{\lambda})}\right)\right]$, with $\log\left(\frac{f(z_{\lambda})}{f(x_{\lambda})}\right) = \log\left|\frac{f(z_{\lambda})}{f(x_{\lambda})}\right| + i \arg\left(\frac{f(z_{\lambda})}{f(x_{\lambda})}\right)$ for $-\pi < \arg\left(\frac{f(z_{\lambda})}{f(x_{\lambda})}\right) \le \pi$. The choice of $\arg(z)$ for $z \in \mathbb{C}$ agrees with that of $\log(z)$ to be employed later in numerical experiments of section. We have an analogous way $s_{\lambda} = \left|\frac{f(z_{\lambda})}{f(x_{\lambda})}\right|^{\frac{1}{m}} \exp\left[\frac{1}{m}\arg\left(\frac{f(z_{\lambda})}{f(x_{\lambda})}\right)\right] = O(e_{\lambda})$. It is important to note that the proposed scheme (5) deduces the well known Ostrowski's method for $m = \alpha = 1$, $\beta = 0$ and $\kappa \to 0$.

In Theorems 1–3, we illustrate that the constructed scheme (5) attains maximum fourth-order of convergence for all $\kappa \neq 0 \in \mathbb{R}$, without adopting any supplementary evaluation of function or its derivative.

Theorem 1. Suppose $x = \gamma$ (say) is a multiple solution of multiplicity m = 2 of function f. Consider that function $f : \mathbb{D} \subset \mathbb{C} \to \mathbb{C}$ is an analytic in \mathbb{D} surrounding the required zero γ . Then, the presented scheme (5) has fourth-order convergence, when

$$\alpha = \beta = \frac{1}{2},\tag{6}$$

and satisfies the following error equation

$$e_{n+1} = -\frac{(2c_1 + \tau_2)}{32} \left(3c_1\tau_2 - 3c_1^2 + 4c_2 + \tau_2^2 \right) e_{\lambda}^4 + O(e_{\lambda}^5), \text{ where } \tau_2 = \kappa f''(\gamma).$$

Proof. Let us consider that $e_{\lambda} = x_{\lambda} - \gamma$ and $c_k = \frac{2!}{(2+k)!} \frac{f^{(2+k)}(\gamma)}{f^{(2)}(\gamma)}$, k = 1, 2, 3, 4 are the error in λ th iteration and asymptotic error constant numbers, respectively. Now, we adopt Taylor's series expansions for the functions $f(x_{\lambda})$ and $f(\mu_{\lambda})$ around $x = \gamma$ with the assumption $f(\gamma) = f'(\gamma) = 0$ and $f''(\gamma) \neq 0$, which are given by

$$f(x_{\lambda}) = \frac{f''(\gamma)}{2!} e_{\lambda}^{2} \left(1 + c_{1}e_{\lambda} + c_{2}e_{\lambda}^{2} + c_{3}e_{\lambda}^{3} + c_{4}e_{\lambda}^{4} + O(e_{\lambda}^{5}) \right)$$
(7)

and

$$f(\mu_{\lambda}) = \frac{f''(\gamma)}{2!} e_{\lambda}^{2} \left[1 + (\tau_{2} + c_{1}) e_{\lambda} + \frac{1}{4} \left(\tau_{2}^{2} + 10\tau_{2}c_{1} + 4c_{2} \right) e_{\lambda}^{2} + \frac{1}{4} \left(5\tau_{2}^{2}c_{1} + 6\tau_{2}c_{1}^{2} + 12\tau_{2}c_{2} + 4c_{3} \right) e_{\lambda}^{3} + \frac{1}{8} \left(\tau_{2}^{3}c_{1} + 14\tau_{2}^{2}c_{1}^{2} + 16\tau_{2}^{2}c_{2} + 28\tau_{2}c_{1}c_{2} + 28\tau_{2}c_{3} + 8c_{4} \right) e_{\lambda}^{4} + O(e_{\lambda}^{5}) \right],$$
(8)

respectively.

By using expressions (7) and (8) in the first substep of scheme (5), we get

$$z_{\lambda} - \gamma = \frac{1}{4} \left(\tau_{2} + 2c_{1} \right) e_{\lambda}^{2} - \left[\frac{1}{16} \tau_{2}^{2} - \frac{1}{2} \tau_{2} c_{1} - c_{2} + \frac{3c_{1}^{2}}{4} \right] e_{\lambda}^{3} + \frac{1}{64} \left[\tau_{2}^{3} - 10c_{1} \left(\tau_{2}^{2} + 16c_{2} \right) - 20\tau_{2}c_{1}^{2} + 64\tau_{2}c_{2} + 72c_{1}^{3} + 96c_{3} \right] e_{\lambda}^{4} + O(e_{\lambda}^{5}).$$

$$(9)$$

Expression (9) and Taylor series expansion, leads us to

$$f(z_{\lambda}) = \frac{f''(\gamma)}{2!} e_{\lambda}^{2} \left[\frac{1}{16} (\tau_{2} + 2c_{1})^{2} e_{\lambda}^{2} - \frac{1}{32} (\tau_{2} + 2c_{1}) \left(\tau_{2}^{2} - 8\tau_{2}c_{1} + 12c_{1}^{2} - 16c_{2} \right) e_{\lambda}^{3} + \frac{1}{256} \left(3\tau_{2}^{4} - 4c_{1} \left(7\tau_{2}^{3} - 48\tau_{2}c_{2} - 96c_{3} \right) + 96\tau_{2}^{2}c_{2} + 32c_{1}^{2}(\tau_{2}^{2} - 32c_{2}) \right) - 80\tau_{2}c_{1}^{3} + 192\tau_{2}c_{3} + 464c_{1}^{4} + 256c_{2}^{2} e_{\lambda}^{4} + O(e_{\lambda}^{5}) \right].$$

$$(10)$$

By adopting the expressions (7), (8) and (10), we obtain

$$s_{\lambda} = \left(\frac{f(z_{\lambda})}{f(x_{\lambda})}\right)^{\frac{1}{m}} = \frac{1}{4} \left(\tau_{2} + 2c_{1}\right) e_{\lambda} + \left(\frac{-\tau_{2}^{3} + 4\tau_{2}^{2}c_{1} - 4\tau_{2}c_{1}^{2} + 16\tau_{2}c_{2} - 32c_{1}^{3} + 32c_{1}c_{2}}{16\left(\tau_{2} + 2c_{1}\right)}\right) e_{\lambda}^{2} + \frac{1}{64} \left(\tau_{2}^{3} - 6\tau_{2}^{2}c_{1} - 22\tau_{2}c_{1}^{2} + 56\tau_{2}c_{2} + 116c_{1}^{3} - 208c_{1}c_{2} + 96c_{3}\right) e_{\lambda}^{3} + O(e_{\lambda}^{4})$$

$$(11)$$

and

$$t_{\lambda} = \left(\frac{f(z_{\lambda})}{f(\mu_{\lambda})}\right)^{\frac{1}{m}} = \frac{1}{4} \left(\tau_{2} + 2c_{1}\right) e_{\lambda} + \left(\frac{-3\tau_{2}^{3} - 4\tau_{2}^{2}c_{1} - 12\tau_{2}c_{1}^{2} + 16\tau_{2}c_{2} - 32c_{1}^{3} + 32c_{1}c_{2}}{16\left(\tau_{2} + 2c_{1}\right)}\right) e_{\lambda}^{2} + \frac{1}{64} \left(7\tau_{2}^{3} - 22\tau_{2}^{2}c_{1} - 14\tau_{2}c_{1}^{2} + 24\tau_{2}c_{2} + 116c_{1}^{3} - 208c_{1}c_{2} + 96c_{3}\right) e_{\lambda}^{3} + O(e_{\lambda}^{4}).$$

$$(12)$$

By using expressions (7)-(12) in the expression (5), we have

$$e_{\lambda+1} = -\frac{1}{4}(\alpha+\beta-1)\left(2c_1+\tau_2\right)e_{\lambda}^2 + \frac{1}{16}\left[(2\beta-1)\tau_2^2 - 2c_1\tau_2(5\alpha+3\beta-4) + 12c_1^2(\alpha+\beta-1) - 16c_2(\alpha+\beta-1)\right]e_{\lambda}^3 + O(e_{\lambda}^4).$$
(13)

In order to obtain fourth-order convergence, the coefficient of e_{λ}^2 and e_{λ}^3 should be zero simultaneously, which deduces

$$lpha + eta - 1 = 0,$$

 $2eta - 1 = 0,$
 $5lpha + 3eta - 4 = 0,$

which further leads us to

$$\alpha = 1 - \beta, \quad \beta = \frac{1}{2}.$$
 (14)

The asymptotic error constant term is obtained if we insert (14) in (13). Then, we have

$$e_{n+1} = -\frac{(2c_1 + \tau_2)}{32} \left(3c_1\tau_2 - 3c_1^2 + 4c_2 + \tau_2^2 \right) e_{\lambda}^4 + O(e_{\lambda}^5).$$
(15)

The expression (15), demonstrates the maximum fourth-order convergence of scheme (5) for m = 2. \Box

Theorem 2. Adopting the same hypotheses of Theorem 1, then the proposed scheme (5) has fourth-order convergence for m = 3 at $\alpha = 1 - \beta$, $\beta \in \mathbb{R}$. It satisfies the following error equation

$$e_{n+1} = \frac{1}{54} \left[3(\beta - 1)\tau_3 + 4b_1^2 - 6b_2 \right] b_1 e_{\lambda}^4 + O(e_{\lambda}^5), \text{ where } \tau_3 = \kappa f'''(\gamma).$$

Proof. We adopt Taylor's series expansions for the functions $f(x_{\lambda})$ and $f(\mu_{\lambda})$ around $x = \gamma$ with the assumption $f(\gamma) = f'(\gamma) = f''(\gamma) = 0$ and $f'''(\gamma) \neq 0$, which are defined as follow:

$$f(x_{\lambda}) = \frac{f'''(\gamma)}{3!} e_{\lambda}^{3} \left(1 + b_{1}e_{\lambda} + b_{2}e_{\lambda}^{2} + b_{3}e_{\lambda}^{3} + b_{4}e_{\lambda}^{4} + O(e_{\lambda}^{5}) \right)$$
(16)

and

$$f(\mu_{\lambda}) = \frac{f'''(\gamma)}{3!} e_{\lambda}^{3} \left[1 + b_{1}e_{\lambda} + \frac{1}{2} \left(\tau_{3} + 2b_{2} \right) e_{\lambda}^{2} + \left(\frac{7}{6} \tau_{3}b_{1} + b_{3} \right) e_{\lambda}^{3} + O(e_{\lambda}^{4}) \right], \tag{17}$$

where $b_j = \frac{3!}{(3+j)!} \frac{f^{(3+j)}(\gamma)}{f^{(3)}(\gamma)}$, j = 1, 2, 3, 4 are asymptotic error constant numbers.

By using expressions (16) and (17) in the first substep of scheme (5), we have

$$z_{\lambda} - \gamma = \frac{b_1}{3}e_{\lambda}^2 + \frac{1}{18}\left(3\tau_3 - 8b_1^2 + 12b_2\right)e_{\lambda}^3 + \left(\frac{1}{9}b_1\left(2\tau_3 - 13b_2\right) + \frac{16b_1^3}{27} + b_3\right)e_{\lambda}^4 + O(e_{\lambda}^5).$$
(18)

Expression (18) and Taylor series expansion, leads us to

$$f(z_{\lambda}) = \frac{f'''(\gamma)}{3!} e_{\lambda}^{3} \left[\frac{b_{1}^{3}}{27} e_{\lambda}^{3} + \frac{1}{54} b_{1}^{2} \left(3\tau_{3} - 8b_{1}^{2} + 12b_{2} \right) e_{\lambda}^{4} + O(e_{\lambda}^{5}) \right].$$
(19)

By adopting the expressions (16), (17) and (19), we obtain

$$t_{\lambda} = \left(\frac{f(z_{\lambda})}{f(\mu_{\lambda})}\right)^{\frac{1}{m}} = \frac{b_1}{3}e_{\lambda} + \frac{1}{18b_1^2}\left(3\tau_3b_1^2 - 10b_1^4 + 12b_2b_1^2\right)e_{\lambda}^2 + \frac{1}{27}\left(3\tau_3b_1 + 23b_1^3 - 48b_2b_1 + 27b_3\right)e_{\lambda}^3 + O(e_{\lambda}^4)$$

$$(20)$$

and

$$s_{\lambda} = \left(\frac{f(z_{\lambda})}{f(x_{\lambda})}\right)^{\frac{1}{m}} = \frac{b_1}{3}e_{\lambda} + \frac{1}{18b_1^2}\left(3\tau_3b_1^2 - 10b_1^4 + 12b_2b_1^2\right)e_{\lambda}^2 + \frac{1}{54}\left(9\tau_3b_1 + 46b_1^3 - 96b_2b_1 + 54b_3\right)e_{\lambda}^3 + O(e_{\lambda}^4).$$
(21)

By using expressions (16)–(21) in the expression (5), we have

$$e_{\lambda+1} = -\frac{1}{3}(\alpha+\beta-1)b_1e_{\lambda}^2 - \frac{(\alpha+\beta-1)}{18}\left(-8b_1^2 + 12b_2 + 3\Delta\kappa\right)e_{\lambda}^3 + O(e_{\lambda}^4).$$
 (22)

In order to obtain fourth-order convergence, the coefficient of e_{λ}^2 and e_{λ}^3 should be zero simultaneously, which deduces

$$\alpha = 1 - \beta. \tag{23}$$

The asymptotic error constant term is obtained if we insert (23) in (22). Then, we have

$$e_{n+1} = \frac{1}{54} \left[3(\beta - 1)\tau_3 + 4c_1^2 - 6c_2 \right] b_1 e_\lambda^4 + O(e_\lambda^5).$$
⁽²⁴⁾

Hence, we proved that scheme (5) has fourth-order convergence for m = 3. \Box

Error for the General Form of the Scheme (5)

Theorem 3. Adopting the same hypotheses of Theorem 2, the scheme given by (5) is of fourth-order convergence for $m \ge 4$. It converges to the following error equation

$$e_{n+1} = \frac{1}{2m^3} \Big[(m+1)s_1^3 - 2ms_1s_2 \Big] e_{\lambda}^4 + O(e_{\lambda}^5).$$

Proof. Let us consider that $e_{\lambda} = x_{\lambda} - \gamma$ and $s_j = \frac{m!}{(m+j)!} \frac{f^{(m+j)}(\gamma)}{f^{(m)}(\gamma)}$, j = 1, 2, 3, 4 are the errors in the λ th iteration and asymptotic error constant numbers, respectively. Now, we adopt Taylor's series expansions for the functions $f(x_{\lambda})$ and $f(\mu_{\lambda})$ around $x = \gamma$ with the assumption $f(\gamma) = f'(\gamma) = f^{(m-1)}(\gamma) = 0$ and $f^{(m)}(\gamma) \neq 0$, which are given by, respectively,

$$f(x_{\lambda}) = \frac{f^{(m)}(\gamma)}{m!} e_{\lambda}^{m} \left(1 + s_{1}e_{\lambda} + s_{2}e_{\lambda}^{2} + s_{3}e_{\lambda}^{3} + s_{4}e_{\lambda}^{4} + O(e_{\lambda}^{5}) \right)$$
(25)

and

$$f(\mu_{\lambda}) = \frac{f^m(\gamma)}{m!} e_{\lambda}^m \left[1 + \sum_{i=0}^2 \Delta_i e_{\lambda}^{i+1} + O(e_{\lambda}^4) \right],$$
(26)

where $\Delta_i = \Delta_i(m, f^{(m)}(\gamma), \kappa, s_1, s_2, s_3, s_4)$. For example, the first coefficient explicitly written as $\Delta_0 = s_1$, $\Delta_1 = s_2$ and $\Delta_2 = \begin{cases} \frac{1}{6} \left(\kappa f^{(4)}(\gamma) + 6\Delta_3 \right), & m = 4 \\ s_3, & m \ge 5 \end{cases}$, etc.

By adopting expressions (25) and (26) in the first substep of scheme (5), we obtain

$$e_{z_{\lambda}} = z_{\lambda} - \gamma = \frac{s_1}{m} e_{\lambda}^2 + \frac{1}{m^2} \left(2ms_2 - (1+m)s_1^2 \right) e_{\lambda}^3 + \frac{1}{m^3} \left(3m^2s_3 + (m+1)^2s_1^3 - m(3m+4)s_2s_1 \right) e_{\lambda}^4 + O(e_{\lambda}^5).$$
(27)

The expression (27) and Taylor series expansion, leads us to

$$f(z_{\lambda}) = \frac{f^{(m)}(\gamma)}{m!} e_{z_{\lambda}}^{m} \left[1 + s_{1}e_{z_{\lambda}} + s_{2}e_{z_{\lambda}}^{2} + s_{3}e_{z_{\lambda}}^{3} + s_{4}e_{z_{\lambda}}^{4} + O(e_{\lambda}^{5}) \right].$$
(28)

From the expressions (25), (26) and (28), we obtain

$$t_{\lambda} = \left(\frac{f(z_{\lambda})}{f(\mu_{\lambda})}\right)^{\frac{1}{m}} = \frac{s_1}{m} e_{\lambda} + \frac{1}{m^2} \left(2ms_2 - (m+2)s_1^2\right) e_{\lambda}^2 + \frac{1}{2m^3} \left((2m^2 + 7m + 7)s_1^3 - 2m(3m + 7)s_1s_2 + 6m^2s_3\right) e_{\lambda}^3 + O(e_{\lambda}^4)$$
(29)

and

$$s_{\lambda} = \left(\frac{f(z_{\lambda})}{f(x_{\lambda})}\right)^{\frac{1}{m}} = \frac{s_{1}}{m}e_{\lambda} + \left(\frac{2}{m}s_{2} - \frac{(m+2)}{m^{2}}s_{1}^{2}\right)e_{\lambda}^{2} + \frac{1}{2m^{3}}\left[(2m^{2} + 7m + 7)s_{1}^{3} - 2m(3m + 7)s_{1}s_{2} + 6m^{2}s_{3}\right]e_{\lambda}^{3} + \frac{1}{3m^{4}}\left[-\left(\frac{11m^{2}}{2} + \frac{33m}{2} + 14\right)s_{1}^{4} - 12m^{2}s_{3}s_{1} + 3m(6m + 11)s_{2}s_{1}^{2}\right] - 6m^{2}s_{2}^{2}\left]e_{\lambda}^{4} + O(e_{\lambda}^{5}).$$
(30)

By using expressions (25)-(30) in the scheme (5), we have

$$e_{\lambda+1} = -\frac{1}{m}(\alpha+\beta-1)b_1e_{\lambda}^2 + (\alpha+\beta-1)\frac{\left((m+1)s_1^2 - 2ms_2\right)}{m^2}e_{\lambda}^3 + O(e_{\lambda}^4).$$
(31)

In order to obtain fourth-order convergence, the coefficient of e_{λ}^2 and e_{λ}^3 should be zero simultaneously, which deduces

$$\alpha = 1 - \beta. \tag{32}$$

The asymptotic error constant term is obtained if we insert (32) in (31). Then, we have

$$e_{n+1} = \frac{1}{2m^3} \Big[(m+1)s_1^3 - 2ms_1s_2 \Big] e_{\lambda}^4 + O(e_{\lambda}^5).$$
(33)

The expression (33) demonstrates maximum fourth-order convergence for all $\kappa \neq 0 \in \mathbb{R}$ and $m \geq 4$, with three different evaluations of function. Hence, our scheme (5) has an optimal convergence order as stated in conjecture given by Kung–Traub. \Box

Remark 1. The final body structure of our proposed scheme (5), is given by

$$z_{\lambda} = x_{\lambda} - m \frac{f(x_{\lambda})}{f[\mu_{\lambda}, x_{\lambda}]},$$

$$x_{\lambda+1} = z_{\lambda} + \frac{(z_{\lambda} - x_{\lambda})(s_{\lambda} + t_{\lambda})}{2(1 - 2s_{\lambda})}.$$
(34)

The above scheme (34) *satisfies all the hypotheses which are mentioned in Theorems* 1-3*.*

Remark 2. It seems from expression (33) (for $m \ge 4$) that β and κ have no importance since they do not exist there. However, they appear in the coefficient of e_{λ}^{5} and the calculation of this term is quite hard, lengthy and time consuming. Moreover, we do not need to mention it here because we already attain the maximum order of convergence. On the other hand, κ can be found in the Equation (15) for m = 2. The β and κ can be found in the expression (24) for m = 3.

Remark 3. It is important to note that the proposed scheme (5) deduces the well known Ostrowski's method for $m = \alpha = 1$, $\beta = 0$ and $\kappa \to 0$.

3. Numerical Experimentation

Efficiency and convergence of our proposed scheme for specializations are as follows. We choose the particular values of disposable parameter κ as $\left(\kappa = \frac{1}{2}\right)$, $\left(\kappa = \frac{1}{4}\right)$ and $\left(\kappa = \frac{1}{10}\right)$ in scheme (34), known by *PM*1, *PM*2 and *PM*3, respectively. The first three examples (1)–(3) are based on real life problems that have multiple roots for their nonlinear models—e.g., Continuous stirred tank reactor (CSTR), Plank's radiation and Van der Waals equation of state. In example (4), we chose a problem of clustering solutions (where roots are close to each others). In the last example (5), we considered a standard academic problem but with higher multiplicity m = 100 in order to check the behavior of our schemes on the problem of larger multiplicity of the required roots.

In Tables 1 and 2, we report our findings for several numbers of significant digits (minimum 3000 significant digits) in order to minimize the rounding off errors. In addition, the CPU timing can be found in Table 3, which are obtained by using the command "AbsoluteTiming[]" in *Mathematica* 9. Due to the limited paper space, we depicted the value up to two significant digits with their exponents. We adopted *Mathematica* 9 with multiple precision arithmetic for calculating the required values. In Tables 1 and 2, $a(\pm b)$ stands for $a \times 10^{(\pm b)}$. The configurations of the used computer are given below:

Processor: Intel(R) Core(TM) i7-4790 CPU @ 3.60 GHz Make: HP RAM: 8:00 GB

System type: 64-bit-Operating System, x64-based processor.

f(x)	λ	KM	SM1	SM2	SK1	SK2	PM1	PM2	<i>PM</i> 3
$f_1(x)$	1	3.1(-3)	3.1(-3)	3.1(-3)	3.1(-3)	3.1(-3)	3.1(-3)	1.6(-3)	7.7(-4)
	2	2.5(-11)	4.2(-11)	2.3(-12)	1.1(-10)	3.2(-11)	1.7(-11)	1.6(-13)	9.7(-15)
	3	6.7(-22)	1.5(-42)	5.9(-24)	2.2(-40)	1.1(-21)	3.3(-22)	1.5(-53)	2.3(-58)
	ρ	1.305	3.999	1.271	3.999	1.309	1.299	4.000	4.000
$f_2(x)$	1	2.3(-6)	2.5(-6)	1.9(-6)	2.8(-6)	1.7(-6)	1.7(-6)	2.1(-6)	2.3(-6)
	2	2.8(-27)	4.6(-27)	1.1(-27)	7.9(-27)	6.8(-28)	6.8(-28)	1.9(-27)	3.0(-27)
	3	6.3(-111)	5.1(-110)	1.5(-112)	5.2(-109)	1.8(-113)	1.8(-113)	1.2(-111)	8.8(-111)
	ρ	4.000	4.000	4.000	4.000	4.000	4.000	4.000	4.000
$f_3(x)$	1	2.2(-2)	3.3(-2)	2.7(-2)	3.7(-2)	1.9(-2)	2.3(-2)	2.3(-2)	2.3(-2)
	2	2.6(-4)	1.5(-3)	6.7(-4)	2.6(-3)	1.1(-4)	3.4(-4)	3.4(-4)	3.4(-4)
	3	1.9(-11)	9.8(-8)	2.1(-9)	1.3(-6)	3.5(-13)	8.6(-11)	9.2(-11)	9.3(-11)
	ρ	3.676	3.115	3.417	2.834	3.822	3.584	3.581	3.580
$f_4(x)$	1	9.0(-4)	3.0(-3)	1.2(-2)	4.7(-3)	2.0(-3)	9.0(-4)	9.0(-4)	9.0(-4)
	2	7.3(-13)	5.2(-10)	1.3(-6)	5.7(-9)	4.9(-11)	7.3(-13)	7.3(-13)	7.3(-13)
	3	3.2(-49)	4.7(-37)	3.4(-12)	1.3(-32)	1.9(-41)	3.2(-49)	3.2(-49)	3.2(-49)
	ρ	4.000	3.999	1.393	3.997	3.999	4.000	4.000	4.000
$f_5(x)$	1	5.3(-5)	2.8(-4)	1.6(-3)	4.9(-4)	1.7(-4)	5.3(-5)	5.3(-5)	5.3(-5)
	2	5.2(-18)	2.9(-14)	1.6(-9)	5.6(-13)	2.3(-15)	5.2(-18)	5.2(-18)	5.2(-18)
	3	5.0(-70)	3.3(-54)	5.2(-18)	9.7(-49)	7.9(-59)	5.0(-70)	5.0(-70)	5.0(-70)
	ρ	4.000	4.000	1.414	4.000	4.000	4.000	4.000	4.000

Table 1. Absolute errors between iterations $(|x_{\lambda+1} - x_{\lambda}|)$ among different iteration functions.

From the numerical results of the examples (4) and (5), it seems that methods *PM*1, *PM*2 and *PM*3 have same numerical results. This is happening because we have mentioned here only two significant digits with exponent. We have found in the computational programming that the digits are different after some places. But, we cannot mention all the digits here due to the page restriction. For instance, the actual results of *PM*1, *PM*2 and *PM*3 in example (4) are 3.177905288(-49), 3.174449760(-49) and 3.172378076(-49) instead of 3.17(-49), 3.17(-49) and 3.17(-49), respectively. Moreover, *PM*1, *PM*2 and *PM*3 have also different CPU timing in these examples and details can be found in Table 3.

Table 2. Contrast on the ground of residual errors (i.e., $|f(x_{\lambda})|$).

f(x)	λ	KM	SM1	SM2	SK1	SK2	PM1	PM2	РМ3
$f_1(x)$	1 2 3	2.0(-5) 1.3(-21) 9.5(-43)	2.0(-5) 3.7(-21) 4.5(-84)	2.0(-5) 1.1(-23) 7.3(-47)	2.0(-5) 2.8(-20) 1.1(-79)	2.0(-5) 2.2(-21) 2.8(-42)	2.0(-5) 6.3(-22) 2.3(-43)	5.4(-6) 5.3(-26) 4.6(-106)	$ \begin{array}{r} 1.3(-6) \\ 2.0(-28) \\ 1.2(-115) \end{array} $
$f_2(x)$	1 2 3		$ \begin{array}{r} 1.1(-19) \\ 6.9(-82) \\ 9.6(-331) \end{array} $	$\begin{array}{r} 4.9(-20) \\ 1.1(-83) \\ 2.3(-338) \end{array}$	$ \begin{array}{r} 1.5(-19) \\ 3.5(-81) \\ 1.0(-327) \end{array} $	$3.5(-20) \\ 2.3(-84) \\ 4.0(-341)$	$3.5(-20) \\ 2.3(-84) \\ 4.0(-341)$	$6.5(-20) \\ 4.7(-83) \\ 1.3(-335)$	
$f_3(x)$	1 2 3	$2.6(-5) \\ 2.0(-9) \\ 1.1(-23)$	$7.5(-5) \\ 6.9(-8) \\ 2.9(-16)$	$\begin{array}{c} 4.5(-5) \\ 1.4(-8) \\ 1.3(-19) \end{array}$	$\begin{array}{c} 1.1(-4) \\ 2.1(-7) \\ 5.2(-14) \end{array}$	$\begin{array}{c} 1.7(-5) \\ 3.7(-10) \\ 3.6(-27) \end{array}$	3.0(-5) 3.4(-9) 2.2(-22)	3.0(-5) 3.6(-9) 2.6(-22)	$3.0(-5) \\ 3.6(-9) \\ 2.6(-22)$
$f_4(x)$	1 2 3	3.0(-56) 4.5(-238) 2.7(-965)	$\begin{array}{c} 1.3(-45) \\ 6.0(-181) \\ 6.0(-722) \end{array}$	$\begin{array}{c} 2.3(-33)\\ 2.3(-113)\\ 1.1(-224)\end{array}$	$\begin{array}{c} 6.6(-42)\\ 3.3(-160)\\ 7.0(-633)\end{array}$	$\begin{array}{c} 2.0(-49) \\ 1.5(-201) \\ 7.5(-810) \end{array}$	3.0(-56) 4.5(-238) 2.7(-965)	3.0(-56) 4.5(-238) 2.7(-965)	$\begin{array}{c} 2.9(-56) \\ 4.4(-238) \\ 2.7(-965) \end{array}$
$f_5(x)$	1 2 3	$\begin{array}{c} 1.3(-380)\\ 4.3(-1681)\\ 5.9(-6883)\end{array}$	$\begin{array}{c} 3.5(-308) \\ 6.7(-1307) \\ 1.1(-5301) \end{array}$	8.2(-232) 2.4(-832) 1.4(-1681)	$\begin{array}{c} 8.1(-284)\\ 5.9(-1178)\\ 2.9(-4754)\end{array}$	$\begin{array}{c} 4.0(-329)\\ 6.4(-1416)\\ 5.1(-5763)\end{array}$	$\begin{array}{c} 1.3(-380)\\ 4.3(-1681)\\ 5.9(-6883)\end{array}$	$\begin{array}{c} 1.3(-380)\\ 4.3(-1681)\\ 5.9(-6883)\end{array}$	$\begin{array}{c} 1.3(-380)\\ 4.3(-1681)\\ 5.9(-6883)\end{array}$

We consider several existing optimal schemes of fourth-order. Firstly, we compare our methods with an optimal fourth-order derivative free scheme given by Kumar et al. [18], in particular we choose expression (27), which is defined as follows:

$$w_{\lambda} = u_{\lambda} - m \frac{f(x_{\lambda})}{f[v_{\lambda}, u_{\lambda}]},$$

$$u_{\lambda+1} = w_{\lambda} - \frac{(m+2)s_{\lambda}}{1-2s_{\lambda}} \frac{f(u_{\lambda})}{f[v_{\lambda}, u_{\lambda}] + 2f[w_{\lambda}, v_{\lambda}]}$$
(35)

where $v_{\lambda} = x_{\lambda} + \beta f(u_{\lambda})$, $s_{\lambda} = \left(\frac{f(w_{\lambda})}{f(u_{\lambda})}\right)^{\frac{1}{m}}$, is denoted by *KM* (for $\beta = 0.5$).

I.M.	Ex. (1)	Ex. (2)	Ex. (3)	Ex. (4)	Ex. (5)	T.T.	A.T.
КM	0.001000	0.071049	0.001000	0.022027	0.029004	0.12408	0.024816
SM1	0.001001	0.056041	0.001000	0.019013	0.028019	0.105074	0.0210148
SM2	0.001000	0.053054	0.001000	0.018013	0.032011	0.105078	0.021014
SK1	0.001000	0.057039	0.001000	0.018013	0.029033	0.106085	0.021217
SK2	0.001001	0.051036	0.001001	0.019025	0.029029	0.101092	0.0202184
PM1	0.001000	0.051036	0.001000	0.017012	0.027019	0.097067	0.0194134
PM2	0.001001	0.050047	0.001001	0.018014	0.026022	0.096085	0.019217
РМ3	0.001000	0.050048	0.001000	0.017024	0.026030	0.095102	0.0190204

Table 3. CPU timing of different iteration functions.

(T.T. and A.T. stand for total time and average time on all five examples, respectively.)

In addition, we compare the same with optimal fourth-order iteration functions developed by Sharma et al. [8], in particular we choose expressions (33) and (35), which are given by

$$z_{\lambda} = t_{\lambda} - m \frac{f(t_{\lambda})}{f[s_{\lambda}, t_{\lambda}]},$$

$$t_{\lambda+1} = z_{\lambda} - \left(mx_{\lambda}y_{\lambda} + mx_{\lambda}^{2} + (m-1)y_{\lambda} + x_{\lambda}\right) \frac{f(t_{\lambda})}{f[s_{\lambda}, t_{\lambda}]}$$
(36)

and

$$z_{\lambda} = t_{\lambda} - m \frac{f(t_{\lambda})}{f[s_{\lambda}, t_{\lambda}]},$$

$$t_{\lambda+1} = z_{\lambda} - \left(\frac{x_{\lambda} - y_{\lambda} + my_{\lambda} - m^{2}x_{\lambda}y_{\lambda} + 2mx_{\lambda}y_{\lambda}}{-mx_{\lambda} + x_{\lambda}^{2} + 1}\right) \frac{f(t_{\lambda})}{f[s_{\lambda}, t_{\lambda}]},$$
(37)

where $s_{\lambda} = t_{\lambda} + \beta f(t_{\lambda})$, $x_{\lambda} = \left(\frac{f(z_{\lambda})}{f(t_{\lambda})}\right)^{\frac{1}{m}}$ and $y_{\lambda} = \left(\frac{f(z_{\lambda})}{f(s_{\lambda})}\right)^{\frac{1}{m}}$. The above expressions are one of their best schemes claimed by them. In particular, for $\beta = 0.5$, we called them *SM*1 and *SM*2, respectively.

Finally, a comparison is given with another optimal fourth-order iterative methods constructed by Kumar et al. [17], in particular we choose methods (M1) and (M3), which are defined by

$$z_{\lambda} = u_{\lambda} - m \frac{f(u_{\lambda})}{f[v_{\lambda}, u_{\lambda}]},$$

$$u_{\lambda+1} = z_{\lambda} - \frac{mh_{\lambda}(3h_{\lambda}+1)}{2} \left(\frac{1}{y_{\lambda}} + 1\right) \frac{f(u_{\lambda})}{f[v_{\lambda}, u_{\lambda}]}$$
(38)

and

$$z_{\lambda} = u_{\lambda} - m \frac{f(u_{\lambda})}{f[v_{\lambda}, u_{\lambda}]},$$

$$u_{\lambda+1} = z_{\lambda} - \frac{mh_{\lambda}(m - 2h_{\lambda})}{2\left(2mh_{\lambda}^{2} - h_{\lambda}(3m + 2) + m\right)} \left(\frac{1}{y_{\lambda}} + 1\right) \frac{f(u_{\lambda})}{f[v_{\lambda}, u_{\lambda}]}$$
(39)

where $v_{\lambda} = u_{\lambda} + \beta f(u_{\lambda})$, $x_{\lambda} = \left(\frac{f(z_{\lambda})}{f(u_{\lambda})}\right)^{\frac{1}{m}}$ and $y_{\lambda} = \left(\frac{f(v_{\lambda})}{f(u_{\lambda})}\right)^{\frac{1}{m}}$, with $h_{\lambda} = \frac{x_{\lambda}}{x_{\lambda}+1}$. The above expressions are one of their best schemes claimed by them. For $\beta = 0.5$, in paticular, we called them *SK*1 and *SK*2, respectively.

Example 1. *Continuous stirred tank reactor (CSTR):*

Here, we assume an isothermal continuous stirred tank reactor (CSTR) problem. Let us consider that components M_1 and M_2 stand for fed rates to the reactors B_1 and $B_2 - B_1$, respectively. Then, we obtain the following reaction scheme in the reactor (for the details see [23]):

$$\begin{split} M_1 + M_2 &\rightarrow B_1 \\ B_1 + M_2 &\rightarrow C_1 \\ C_1 + M_2 &\rightarrow D_1 \\ C_1 + M_2 &\rightarrow E_1 \end{split}$$

Douglas [24] *studied the above model, when he was designing a simple model for feedback control systems. He converted the above model in to the following mathematical expression:*

$$R_{C_1} \frac{2.98(x+2.25)}{(x+1.45)(x+2.85)^2(x+4.35)} = -1,$$

where R_{C_1} is the gain of proportional controller. Expression (40) is balanced for the negative real values of values of R_{C_1} . In particular, by choosing $R_{C_1} = 0$, we yield

$$f_1(x) = x^4 + 11.50x^3 + 47.49x^2 + 83.06325x + 51.23266875.$$
(40)

The zeros of function f_1 are known as the poles of the open-loop transfer function. The function f_1 has four zeros $\gamma = -1.45, -2.85, -2.85, -4.35$. However, our desired one is $\gamma = -2.85$ with multiplicity m = 2. We assume $x_0 = -2.8$ as the starting point for f_1 .

The results obtained from the Tables 1 and 2 conclude that our methods perform far better than the existing ones in the terms of difference between two iterations, residual error and computational order of convergence in the case of example 1.

Example 2. *Planck's radiation problem:*

Consider the Planck's radiation equation that determines the spectral density of electromagnetic radiations released by a black-body at a given temperature, and at thermal equilibrium [25] *as*

$$G(y) = \frac{8\pi chy^{-5}}{e^{\frac{ch}{ykT}} - 1},$$

where T, y, k, h, and c denote the absolute temperature of the black-body, wavelength of radiation, Boltzmann constant, Planck's constant, and speed of light in the medium (vacuum), respectively. To evaluate the wavelength y, which results in the maximum energy density G(y), set G'(y) = 0. We obtained the following equation

$$\frac{\left(\frac{ch}{ykT}\right)e^{\frac{ch}{ykT}}}{e^{\frac{ch}{ykT}}-1} = 5.$$

Further, the nonlinear equation is formulated by setting $x = \frac{ch}{ykT}$ *as follows:*

$$f_2(x) = \left(e^{-x} - 1 + \frac{x}{5}\right)^3.$$

The exact root is $\gamma = 4.96511423174428$ of multiplicity m = 3 and with this root one can easily find the wave length y form the relation $x = \frac{ch}{ykT}$. The Planck's problem is tested with initial guess $x_0 = 5.4$ and computational results are depicted in Tables 1 and 2. **Example 3.** Van der Waals equation of state:

$$\left(P+\frac{a_1n^2}{V^2}\right)\left(V-na_2\right)=nRT,$$

describes the nature of a real gas between two gases, namely a_1 and a_2 , when we introduce the ideal gas equations. For calculating the volume V of gases, we need the solution of preceding expression in terms of remaining constants

$$PV^{3} - (na_{2}P + nRT)V^{2} + \alpha_{1}n^{2}V - \alpha_{1}\alpha_{2}n^{2} = 0.$$

For choosing the particular values of gases α_1 and α_2 , we can easily obtain the values for *n*, *P* and *T*. Then, we yield

$$f_3(x) = x^3 - 5.22x^2 + 9.0825x - 5.2675.$$

The function f_3 has three zeros, and among them $\gamma = 1.75$ is a multiple zero of multiplicity m = 2and $\gamma = 1.72$ is a simple zero. We choose the starting guess $x_0 = 2$ for the required zero $\xi = 1.75$ in f_3 . The computational results are given in Tables 1 and 2.

Example 4. We picked a root clustering problem (the details can be found in Zeng [26]):

$$f_4(x) = (x-2)^{15}(x-4)^5(x-3)^{10}(x-1)^{20}.$$
(41)

In the above function, we have four multiple zeros x = 1, 2, 3 and 4 and multiplicity of the corresponding zero is 20, 15, 10 and 5, respectively. All multiple zeroes are quite close to each other. We chose x = 1 multiple zero of multiplicity 20 for the computational point of view. The computational results are depicted in Tables 1 and 2, based on the initial approximation $x_0 = 0.8$.

Example 5. We considered another problem in order to test the efficiency of our methods on higher-order multiplicity of the required root, which is given as follows:

$$f_5(x) = \left((x-1)^3 - 1 \right)^m, \ m = 100.$$
 (42)

Equation (42) has a zero x = 2. of multiplicity 100. The computational results are mentioned in Tables 1 and 2, based on the starting point $x_0 = 2.1$.

4. Concluding Remarks

- This paper suggest a new fourth-order derivative free family of Ostrowski's method for multiple roots of nonlinear equations.
- There are no doubts that a scheme with weight function has a more general form, such as

 (2) and (3). However, finding a weight function with some pre-defined conditions is not an
 easy task. With the help of free disposable parameters, we can easily construct several new
 iterative schemes. So, by changing the parameter *κ* in our scheme, we can obtain several new
 derivative-free optimal methods.
- It is worthy to note that our scheme (34), consumes only three different values of the function—i.e., *f*(*x*_λ), *f*(*μ*_λ) and *f*(*z*_λ), at each iteration, without any derivative. Thus, our scheme (34) has optimal fourth-order convergence in sense of the classic Kung–Traub conjecture.
- Our methods have simple body structure as compared to the existing ones.
- We conclude from the numerical experimentation that our methods, *PM*1, *PM*2 and *PM*3, have at least equal performance as compared to *KM* but also demonstrate better behavior than *SM*1, *SM*2, *SK*1 and *SK*2 in examples 1, 2, 4 and 5. In addition, our methods illustrate far better

performance than *KM* in terms of residual errors and errors among two consecutive iterations (the results can be seen in Tables 1 and 2) in the case of example 1.

- Our methods consume the lowest CPU time (total and average time on the examples) as compared to all other depicted methods.
- Based on the results in Tables 1–3, our methods are more effective and a better alternative to the earlier existing schemes.

Author Contributions: R.B.: Conceptualization; Methodology; Validation; Writing—Original Draft Preparation; Writing—Review and Editing. M.S., F.O.M. and S.K.A.: Review and Editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

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

References

- 1. Le, D. An efficient derivative free method for solving nonlinear equations. *ACM Trans. Math. Soft.* **1985**, *11*, 250–262. [CrossRef]
- 2. Hasan, M.A. Derivative-free family of higher order root finding methods. In Proceedings of the 2009 American Control Conference, St. Louis, MO, USA, 10–12 June 2009; pp. 5351–5356. [CrossRef]
- 3. Cordero, A.; Hueso, J.L.; Martinez, E.; Torregrosa, J.R. A new technique to obtain derivative-free optimal iterative methods for solving nonlinear equations. *J. Comput. Appl. Math.* **2013**, *252*, 95–102. [CrossRef]
- 4. Zhanlav, T.; Otgondorj, K. Comparison of some optimal derivative free three point iterations. *Numer. Anal. Approx. Theory* **2020**, *49*, 76–90.
- Behl, R.; Motsa, S.S.; Kansal, M.; Kanwar, V. Fourth-Order Derivative-Free Optimal Families of King's and Ostrowski's Methods. In *Mathematical Analysis and Its Applications*; Agrawal, P., Mohapatra, R., Singh, U., Srivastava, H., Eds.; Springer Proceedings in Mathematics & Statistics 143; Springer: New Delhi, India, 2015.
- 6. Cordero, A.B.; Torregrosa, J.R.S. Low-complexity root-finding iteration functions with no derivatives of any order of convergence. *J. Comput. Appl. Math.* **2015**, *275*, 502–515. [CrossRef]
- Hueso, J.L.; Martńez, E.; Teruel, C. Determination of multiple roots of nonlinear equations and applications. *Math. Chem.* 2015, 53, 880–892. [CrossRef]
- 8. Sharma, J.R.; Kumar, S.; Jntschi, L. On a class of optimal fourth order multiple root solvers without using derivatives. *Symmetry* **2019**, *11*, 452. [CrossRef]
- 9. Behl, R.; Salimi, M.; Ferrara, M.; Sharifi, S.; Samaher, K.A. Some real life applications of a newly constructed derivative free iterative scheme. *Symmetry* **2019**, *11*, 239. [CrossRef]
- 10. Salimi, M.; Lotfi, T.; Sharifi, S.; Siegmund, S. Optimal Newton-Secant like methods without memory for solving nonlinear equations with its dynamics. *Int. J. Comput. Math.* **2017**, *94*, 1759–1777. [CrossRef]
- 11. Salimi, M.; Nik Long, N.M.A.; Sharifi, S.; Pansera, B.A. A multi-point iterative method for solving nonlinear equations with optimal order of convergence. *Jpn. J. Ind. Appl. Math.* **2018**, *35*, 497–509. [CrossRef]
- 12. Matthies, G.; Salimi, M.; Sharifi, S.; Varona, J.L. An optimal eighth-order iterative method with its dynamics. *Jpn. J. Ind. Appl. Math.* **2016**, *33*, 751–766. [CrossRef]
- 13. Sharifi, S.; Ferrara, M.; Salimi, M.; Siegmund, S. New modification of Maheshwari method with optimal eighth order of convergence for solving nonlinear equations. *Open Math. Formerly Cent. Eur. J. Math.* **2016**, *14*, 443–451.
- 14. Lotfi, T.; Sharifi, S.; Salimi, M.; Siegmund, S. A new class of three point methods with optimal convergence order eight and its dynamics. *Numer. Algor.* **2016**, *68*, 261–288. [CrossRef]
- 15. Traub, J.F. *Iterative Methods for the Solution of Equations*; Prentice-Hall Series in Automatic Computation: Englewood Cliffs, NJ, USA, 1964.
- 16. Ortega, J.M.; Rheinboldt, W.C. *Iterative Solution of Nonlinear Equations in Several Variables*; Academic Press: New York, NY, USA, 1970.
- 17. Sharma, J.R.; Kumar, S.; Jntschi, L. On Derivative Free Multiple-Root Finders with Optimal Fourth Order Convergence. *Mathematics* 2020, *8*, 1091. [CrossRef]

- 18. Kumar, S.; Kumar, D.; Sharma, J.R.; Cesarano, C.; Aggarwal, P.; Chu, Y.M. An optimal fourth order derivative-free Numerical Algorithm for multiple roots. *Symmetry* **2020**, *12*, 1038. [CrossRef]
- 19. Kung, H.T.; Traub, J.F. Optimal order of one-point and multipoint iteration. *J. Assoc. Comput. Mach.* **1974**, 21, 643–651. [CrossRef]
- 20. Ostrowski, A.M. Solutions of Equations and System of Equations; Academic Press: New York, NY, USA, 1964.
- 21. Petković, M.S.; Neta, B.; Petković, B.L.D.; Džunić, J. *Multipoint Methods for Solving Nonlinear Equations*; Academic Press: New York, NY, USA, 2012.
- 22. Ahlfors, L.V.; Complex Analysis; McGraw-Hill Book, Inc.:, NY, USA, 1979.
- 23. Constantinides, A.; Mostoufi, N. *Numerical Methods for Chemical Engineers with MATLAB Applications;* Prentice Hall PTR: Englewood Cliffs, NJ, USA, 1999.
- 24. Douglas, J.M. Process Dynamics and Control; Prentice Hall: Englewood Cliffs, NJ, USA, 1972; Volume 2.
- 25. Bradie, B. A Friendly Introduction to Numerical Analysis; Pearson Education Inc.: New Delhi, India, 2006.
- 26. Zeng, Z. Computing multiple roots of inexact polynomials. Math. Comput. 2004, 74, 869–903. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



 \odot 2020 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 (http://creativecommons.org/licenses/by/4.0/).