Applied Mathematics and Computation 222 (2013) 564–574

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

New optimal class of higher-order methods for multiple roots, permitting $f'(x_n) = 0$

V. Kanwar *, Saurabh Bhatia, Munish Kansal

University Institute of Engineering and Technology, Panjab University, Chandigarh 160 014, India

ARTICLE INFO

Keywords:
Nonlinear equations
Multiple roots
Newton's method
Optimal order of convergence
Efficiency index

ABSTRACT

Finding multiple zeros of nonlinear functions pose many difficulties for many of the iterative methods. A major difficulty in the application of iterative methods is the selection of initial guess such that neither guess is far from zero nor the derivative is small in the vicinity of the required root, otherwise the methods would fail miserably. Finding a criterion for choosing initial guess is quite cumbersome and therefore, more effective globally convergent algorithms for multiple roots are still needed. Therefore, the aim of this paper is to present an improved optimal class of higher-order methods having quartic convergence, permitting $f'(x) = 0$ in the vicinity of the required root. The present approach of deriving this optimal class is based on weight function approach. All the methods considered here are found to be more effective and comparable to the similar robust methods available in literature.

1. Introduction

Finding the multiple roots of nonlinear equations efficiently and accurately, is a very interesting and challenging problem in computational mathematics. It has many applications in engineering and other applied sciences. We consider an equation of the form

$$f(x) = 0, \tag{1.1}$$

where $f : D \subset \mathbb{R} \to \mathbb{R}$ be a nonlinear continuous function on D. Analytical methods for solving such equations are almost non-existent and therefore, it is only possible to obtain approximate solutions by relying on numerical methods based on iterative procedures. So, in this paper, we concern ourselves with iterative methods to find the multiple root r_m with multiplicity $m > 1$ of a nonlinear Eq. (1.1), i.e. $f^i(r_m) = 0$, $i = 0, 1, 2, 3, \ldots, m-1$ and $f^m(r_m) \neq 0$ (a condition for $x = r_m$ to be a root of multiplicity m). These multiple roots pose difficulties for root-finding methods as function does not change sign at even multiple roots, precluding the use of bracketing methods, limiting one to open methods.

Modified Newton's method [1]

$$x_{n+1} = x_n - m\frac{f(x_n)}{f'(x_n)}, \quad n \geqslant 0 \tag{1.2}$$

is an important and basic method for finding multiple roots of nonlinear Eq. (1.1). It is probably the best known and most widely used algorithm for solving such problems. It converges quadratically and requires the prior knowledge of multiplicity

* Corresponding author.
E-mail address: vmithil@yahoo.co.in (V. Kanwar).

0096-3003/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.amc.2013.06.097

m. However, a major difficulty in the application of modified Newton's method is the selection of initial guess such that neither guess is far from zero nor the derivative is small in the vicinity of the required root, otherwise the method fails miserably. Finding a criterion for choosing initial guess is quite cumbersome and therefore, more effective globally convergent algorithms are still needed. Furthermore, inflection points on the curve, with in the region of search, are also trouble some and may cause the search to diverge or converge to undesired root. In order to overcome these problems, we consider the following modified one-point iterative scheme

$$x_{n+1} = x_n - m\frac{f(x_n)}{f'(x_n) - pf(x_n)}. \tag{1.3}$$

In order to obtain quadratic convergence, the entity in the denominator should be largest in magnitude. For $p = 0$ and $m = 1$, we obtain the classical Newton's method. The error equation of scheme (1.3) is given by

$$e_{n+1} = \left(\frac{-p + c_1}{m}\right)e_n^2 + O(e_n^3), \tag{1.4}$$

where $e_n = x_n - r_m,\ c_k = \frac{m!}{k!}\frac{f^{(k)}(r_m)}{f^{(m)}(r_m)},\ k = 2, 3, \ldots$

This work is an extension of the one-point modified family of Newton's method [2,3] for simple roots. Recently, Kumar et al. [4] have also derived this family of Newton's method geometrically by implementing approximation through a straight line. They have proved that for small values of p, slope or angle of inclination of straight line with *x–axis* becomes smaller, i.e. as $p \to 0$, the straight line tends to *x–axis*. This means that next approximation will move faster towards the desired root.

As the order of an iterative method increases, so does the number of functional evaluations per step. The efficiency index [5,6] gives a measure of the balance between those quantities, according to the formula $p^{\frac{1}{d}}$, where p is the order of convergence of the method and d the number of functional evaluations per step. According to the Kung–Traub conjecture [7], the order of convergence of any multipoint method cannot exceed the bound 2^{n-1}, called the optimal order. Nowadays, obtaining an optimal multipoint method for multiple roots having quartic convergence and converges to the required root even though the guess is far from zero or the derivative is small in the vicinity of the required root is an open and challenging problem in computational mathematics. But till the date, we do not have any optimal method of order-four that can overcome these problems, in the case of multiple roots.

The contents of this paper unfold the material in what follows. Section 2 presents a brief look at the existing multipoint families of higher-order methods for multiple roots, where it is followed by Section 3 wherein our main contribution lie. We develop a general class of higher-order methods, which will converge in case the initial guess is far from zero or the derivative is small in the vicinity of the required root. Some new families of higher-order methods are also proposed. In Section 4, we have proved the order of convergence of our proposed scheme. Section 5 includes a numerical comparison between proposed methods without memory and the existing robust methods available in literature and finally, the concluding remarks of the paper have been drawn.

2. Brief literature review

In recent years, some modifications of Newton's method for multiple roots have been proposed and analyzed by Kumar et al. [8], Li et al. [9,10], Neta and Johnson [11], Sharma and Sharma [12], Zhou et al. [13], and the references cited therein. There are, however, not yet so many fourth or higher-order methods known that can handle the case of multiple roots.

In [11], Neta and Johnson have proposed a fourth-order method requiring one-function and three derivative evaluations per iteration. This method is based on Jarratt's method [14] given by the iteration function

$$x_{n+1} = x_n - \frac{f(x_n)}{a_1 f'(x_n) + a_2 f'(y_n) + a_3 f'(\eta_n)}, \tag{2.1}$$

where

$$\begin{cases} u_n = \frac{f(x_n)}{f'(x_n)}, \\ y_n = x_n - a u_n, \\ v_n = \frac{f(x_n)}{f'(y_n)}, \\ \eta_n = x_n - b u_n - c v_n. \end{cases}$$

Neta and Johnson [11] gave a table of values for the parameters a, b, c, a_1, a_2, a_3 for several values of m. But they do not give a closed formula for general case.

Inspired by the work of Jarratt [14], Sharma and Sharma [12] present the following optimal variant of Jarratt's method given by

$$\begin{cases} y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \frac{m}{8}\left[(m^3 - 4m + 8) - (m+2)^2\left(\frac{m}{m+2}\right)^m \frac{f'(x_n)}{f'(y_n)} \times \left(2(m-1) - (m+2)\left(\frac{m}{m+2}\right)^m \frac{f'(x_n)}{f'(y_n)}\right)\frac{f'(x_n)}{f'(y_n)}\right]. \end{cases} \tag{2.2}$$

More recently, Zhou et al. [13] have developed many fourth-order multipoint methods by considering the following iterative scheme:

$$\begin{cases} y_n = x_n - t\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} Q\left(\frac{f'(y_n)}{f'(x_n)}\right), \end{cases} \tag{2.3}$$

where

$$\begin{cases} t = \frac{2m}{m+2}, \\ Q(u) = m, \\ Q'(u) = \frac{-1}{4} m^{3-m}(m+2)^m, \\ Q''(u) = \frac{1}{4} m^4 \left(\frac{m}{m+2}\right)^{-2m}, \end{cases} \tag{2.4}$$

and $u = \left(\frac{m}{m+2}\right)^{m-1}$.

However, all these multipoint methods are the variants of Newton's method and the iteration can be aborted due to the overflow or leads to divergence, if the derivative of the function at an iterative point is singular or almost singular, which restrict their applications in practical.

Therefore, construction of an optimal multipoint method having quartic convergence and converge to the required root even though the guess is far from zero or the derivative is small in the vicinity of the required root is an open and challenging problem in computational mathematics. With this aim, we intend to propose an optimal scheme of higher-order methods in which $f'(x) = 0$ is permitted at some points in the neighborhood of required root. The present approach of deriving this optimal class of higher-order methods is based on weight function approach. All the proposed methods considered here are found to be more effective and comparable to the existing robust methods available in literature.

3. Construction of novel techniques without memory

In this section, we intend to develop a new modified optimal class of higher-order methods for multiple roots, which will converge even though $f'(x) = 0$, is permitted at some point. For this purpose, we consider the following two-step scheme as follows:

$$\begin{cases} y_n = x_n - \frac{2m}{m+2} \frac{f(x_n)}{f'(x_n) - pf(x_n)}, \\ x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - pf(x_n)} Q\left(\frac{f'(y_n) + hf'(x_n)}{tf'(x_n) - pf(x_n)}\right), \end{cases} \tag{3.1}$$

where p, t and h are three free disposable parameters and $Q\left(\frac{f'(y_n) + hf'(x_n)}{tf'(x_n) - pf(x_n)}\right)$ is a real-valued weight function such that the order of convergence reaches at the optimal level four without using any more functional evaluations. Theorem 4.1 indicates that under what conditions on the disposable parameters in (3.1), the order of convergence will reach at the optimal level four.

4. Order of convergence

Theorem 4.1. *Let $f : D \subseteq \mathbb{R} \to \mathbb{R}$ be a sufficiently smooth function defined on an open interval D, enclosing a multiple zero of f(x), say $x = r_m$ with multiplicity $m > 1$. Then the family of iterative methods defined by (3.1) has fourth-order convergence when*

$$\begin{cases} t = \frac{1}{m+1}, \\ h = -\left(\frac{m}{2+m}\right)^m, \\ Q(\mu) = m, \\ Q'(\mu) = -\frac{m^3\left(\frac{m}{2+m}\right)^{-m}}{4(1+m)}, \\ Q''(\mu) = \frac{m^4\left(\frac{m}{2+m}\right)^{-2m}}{4(1+m)^2}, \\ |Q'''(\mu)| < \infty, \end{cases} \tag{4.1}$$

where $\mu = \left(\frac{m}{m+2}\right)^{m-1}$ and it satisfies the following error equation

$$\begin{aligned} e_{n+1} = \frac{(2+m)^{-3m}}{6m^{10}} \Big[& 3(-64Q'''(\mu)m^{3m}(1+m)^3 - m^5(2+m)^{3m}(24 - 4m + 4m^2 + 3m^3 + m^4))pc_1^2 \\ & + 2(32Q'''(\mu)m^{3m}(1+m)^3 + m^5(2+m)^{3m}(12 - 2m + 2m^2 + 2m^3 + m^4))c_1^3 + 6c_1(32Q'''(\mu)m^{3m}(1+m)^3 p^2 \\ & - m^5(2+m)^{3m}((-12 + 2m - 2m^2 - m^3)p^2 + m^4 c_2)) + \frac{1}{(2+m)^2}\{-64Q'''(\mu)m^{3m}(1+m)^3(2+m)^2 p^3 \\ & + m^5(2+m)^{3m}((2+m)^2 p(-24 + 4m - 4m^2 - m^3 + m^4)p^2 + 6m^4 c_2) + 6m^6 c_3)\} \Big] e_n^4 + O(e_n)^5, \end{aligned} \tag{4.2}$$

where e_n and c_k are already defined in Eq. (1.4).

Proof. Let $x = r_m$ be a multiple zero of $f(x)$. Expanding $f(x_n)$ and $f'(x_n)$ about $x = r_m$ by the Taylor's series expansion, we have

$$f(x_n) = \frac{f^{(m)}(r_m)}{m!} e_n^m \left(1 + c_1 e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4\right) + O(e_n^5), \tag{4.3}$$

and

$$f'(x_n) = \frac{f^{(m-1)}(r_m)}{(m-1)!} e_n^{m-1} \left(1 + \frac{m+1}{m} c_1 e_n + \frac{m+2}{m} c_2 e_n^2 + \frac{m+3}{m} c_3 e_n^3 + \frac{(m+4)}{m} c_4 e_n^4\right) + O(e_n^5), \tag{4.4}$$

respectively. □

From Eqs. (4.3) and (4.4), we have

$$\begin{aligned}\frac{f(x_n)}{f'(x_n) - pf(x_n)} &= \frac{e_n}{m} + \frac{(p - c_1)e_n^2}{m^2} + \frac{(p^2 - 2pc_1 + (1+m)c_1^2 - 2mc_2)e_n^3}{m^3} \\ &\quad + \frac{\left(p^3 + (3+2m)pc_1^2 - (1+m)^2 c_1^3 - 4mpc_2 + c_1(-3p^2 + m(4+3m)c_2) - 3m^2 c_3\right)e_n^4}{m^4} + O(e_n)^5,\end{aligned} \tag{4.5}$$

and in the combination of Taylor series expansion of $f'\left(x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n) - pf(x_n)}\right)$ about $x = r_m$, we have

$$\begin{aligned}f'(y_n) &= f'\left(x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n) - pf(x_n)}\right) \\ &= f^{(m)}(r_m)e_n^{m-1}\left(\frac{\left(\frac{m}{2+m}\right)^m (2+m)}{m!} + \frac{\left(\frac{m}{2+m}\right)^m (-2(-2+m+m^2)p + (-4+2m+3m^2+m^3)c_1)e_n}{m^2 m!}\right. \\ &\quad \left. + \frac{\left(\frac{m}{2+m}\right)^m (-4(-2+m+m^2)p^2 - 2(8-4m-4m^2+m^3+m^4)pc_1 - 4(-2+m)c_1^2 + m^2(-8+4m+4m^2+m^3)c_2)e_n^2}{m^4 m!} + O(e_n^3)\right).\end{aligned} \tag{4.6}$$

Furthermore, we have

$$\begin{aligned}\frac{f'(y_n) + hf'(x_n)}{tf'(x_n) - pf(x_n)} &= \frac{hm + \left(\frac{m}{2+m}\right)^m (2+m)}{mt} + \frac{\left(p\left(hm^2 - \left(\frac{m}{2+m}\right)^m (2+m)(-2t + m(-1+2t))\right) - 4\left(\frac{m}{2+m}\right)^m tc_1\right)e_n}{m^3 t^2} \\ &\quad + \frac{1}{m^5 t^3}\left(-p^2\left(-hm^3 + \left(\frac{m}{2+m}\right)^m (2+m)(-4t^2 + m^2(-1+2t) + 2mt(-1+2t))\right)\right. \\ &\quad + pt\left(-hm^3 + \left(\frac{m}{2+m}\right)^m (4m(-1+t) - 16t + m^3(-1+2t) + m^2(-2+6t))\right)c_1 \\ &\quad \left. + 4\left(\frac{m}{2+m}\right)^m (2+m^2)t^2 c_1^2 - 8m^2\left(\frac{m}{2+m}\right)^m t^2 c_2\right)e_n^2 + O(e_n)^3.\end{aligned} \tag{4.7}$$

Since it is clear from (4.7) that $\left(\frac{f'(y_n)+hf'(x_n)}{tf'(x_n)-pf(x_n)}\right) - \mu$ is of order e_n, where $\mu = \frac{hm + \left(\frac{m}{2+m}\right)^m (2+m)}{mt}$. Hence, we can consider the Taylor's expansion of the weight function Q in the neighborhood of μ. Therefore, we have

$$\begin{aligned}Q\left(\frac{f'(y_n) + hf'(x_n)}{tf'(x_n) - pf(x_n)}\right) &= Q(\mu) + \left(\frac{f'(y_n) + hf'(x_n)}{tf'(x_n) - pf(x_n)}\right)Q'(\mu) + \frac{1}{2!}\left(\frac{f'(y_n) + hf'(x_n)}{tf'(x_n) - pf(x_n)}\right)^2 Q''(\mu) \\ &\quad + \frac{1}{3!}\left(\frac{f'(y_n) + hf'(x_n)}{tf'(x_n) - pf(x_n)}\right)^3 Q'''(\mu) + O(e_n^4).\end{aligned} \tag{4.8}$$

Using 4.5, 4.7 and 4.8 in the scheme (3.1), we have the following error equation

$$\begin{aligned}e_{n+1} &= e_n - \frac{f(x_n)}{f'(x_n) - pf(x_n)} Q\left(\frac{f'(y_n) + hf'(x_n)}{tf'(x_n) - pf(x_n)}\right) = \left(1 - \frac{Q(\mu)}{m}\right)e_n \\ &\quad + \left(\frac{Q(\mu)(-p + c_1)}{m^2} + \frac{Q'(\mu)\left(p\left(-hm^2 + \left(\frac{m}{2+m}\right)^m (2+m)(-2t + m(-1+2t))\right) + 4\left(\frac{m}{2+m}\right)^m tc_1\right)}{m^4 t^2}\right)e_n^2 \\ &\quad + \frac{1}{m^5}\left(A_1 - \frac{1}{2m^2 t^4}\left(A_2 + 2Q'(\mu)mtA_3\right)\right)e_n^3 + O(e_n^4),\end{aligned} \tag{4.9}$$

where

$$A_1 = \frac{Q'(\mu)(2+m)^{-m}(p-c_1)(-hm^2(2+m)^m p + m^m((2+m)p(-m+2(-1+m)t)+4tc_1))}{t^2} - Q(\mu)m^2(p^2-2pc_1+(1+m)c_1^2-2mc_2),$$

$$A_2 = Q''(\mu)\left(p(hm^2+m^m(2+m)^{1-m}(m+2t-2mt)) - 4\left(\frac{m}{2+m}\right)^m tc_1\right)^2,$$

$$A_3 = p^2(hm^3(2+m)^m - m^m(2+m)(-m^2+2(-1+m)mt+4(-1+m)t^2)) - hm^3(2+m)^m ptc_1 + m^m t(c_1(-m(4+m(2+m))p+2(-8+m(1+m)(2+m))pt+4(2+m^2)tc_1) - 8m^2tc_2).$$

For obtaining an optimal general class of fourth-order iterative methods, the coefficients of e_n, e_n^2, and e_n^3 in the error Eq. (4.9) must be zero simultaneously. After simplifying the Eq. (4.9), we have the following equations involving of $Q(\mu)$, $Q'(\mu)$, and $Q''(\mu)$

$$\begin{cases} Q(\mu) = m, \\ \frac{Q(\mu)(-p+c_1)}{m^2} = -\frac{Q'(\mu)\left(p\left(-hm^2+\left(\frac{m}{2+m}\right)^m(2+m)(-2t+m(-1+2t))\right)+4\left(\frac{m}{2+m}\right)^m tc_1\right)}{m^4t^2}, \\ A_1 = 0, \\ A_2 = 0, \\ A_3 = 0, \end{cases} \tag{4.10}$$

respectively.

Solving the above equations for $Q(\mu)$, $Q'(\mu)$, $Q''(\mu)$, t, and h, we get

$$\begin{cases} t = \frac{1}{m+1}, \\ h = -\left(\frac{m}{2+m}\right)^m, \\ Q(\mu) = m, \\ Q'(\mu) = -\frac{m^3\left(\frac{m}{2+m}\right)^{-m}}{4(1+m)}, \\ Q''(\mu) = \frac{m^4\left(\frac{m}{2+m}\right)^{-2m}}{4(1+m)^2}, \end{cases} \tag{4.11}$$

where $\mu = \frac{hm+\left(\frac{m}{2+m}\right)^m(2+m)}{mt}$.

After using the recently obtained values of $t = \frac{1}{m+1}$ and $h = -\left(\frac{m}{2+m}\right)^m$ in $\mu = \frac{hm+\left(\frac{m}{2+m}\right)^m(2+m)}{mt}$, we further get $\mu = \left(\frac{m}{m+2}\right)^{m-1}$.

Using the above conditions, the scheme (3.1) will satisfy the following error equation

$$e_{n+1} = \frac{(2+m)^{-3m}}{6m^{10}}\Big[3(-64Q'''(\mu)m^{3m}(1+m)^3 - m^5(2+m)^{3m}(24-4m+4m^2+3m^3+m^4))pc_1^2 + 2(32Q'''(\mu)m^{3m}(1+m)^3 + m^5(2+m)^{3m}(12-2m+2m^2+2m^3+m^4))c_1^3 + 6c_1(32Q'''(\mu)m^{3m}(1+m)^3p^2 - m^5(2+m)^{3m}((-12+2m-2m^2-m^3)p^2+m^4c_2)) + \frac{1}{(2+m)^2}\{-64Q'''(\mu)m^{3m}(1+m)^3(2+m)^2p^3 + m^5(2+m)^{3m}((2+m)^2p(-24+4m-4m^2-m^3+m^4)p^2+6m^4c_2)+6m^6c_3)\}\Big]e_n^4 + O(e_n)^5, \tag{4.12}$$

where $|Q'''(\mu)| < \infty$ and $p \in \mathbb{R}$ is a free disposable parameter.

This reveals that the general two-step class of higher-order methods (3.1) reaches the optimal order of convergence four by using only three functional evaluations per full iteration. The beauty of our proposed optimal general class is that it will converge to the required root even $f'(x) = 0$ unlike Jarratt's method and existing robust methods. This completes the proof of the Theorem 4.1.

***Note:* Selection of parameter '*p*' in family (3.1)**

The parameter 'p' in family (3.1) is chosen so as to give the largest value of denominator. In order to make this happen, we take

$$p = \begin{cases} +ve, & \text{if } f(x_n)f'(x_n) \leqslant 0, \\ -ve, & \text{if } f(x_n)f'(x_n) \geqslant 0. \end{cases} \tag{4.13}$$

5. Some special cases

Finally, by using specific values of t and h, which are defined in Theorem 4.1, we get the following general class of higher-order iterative methods given by

$$\begin{cases} y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)-pf(x_n)}, \\ x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)-pf(x_n)} Q\left(\frac{(m+1)\left(f'(y_n)-\left(\frac{m}{m+2}\right)^m f'(x_n)\right)}{f'(x_n)-(m+1)pf(x_n)}\right), \end{cases} \tag{5.1}$$

where $Q\left(\frac{(m+1)\left(f'(y_n)-\left(\frac{m}{m+2}\right)^m f'(x_n)\right)}{f'(x_n)-(m+1)pf(x_n)}\right)$ is a weight function which satisfies the conditions defined in Theorem 4.1. Now, we shall consider some particular cases of the proposed scheme (5.1) depending upon the weight function $Q(x)$ and p as follow:

Case 1. Let us consider the following weight function

$$Q(x) = \frac{m(1+m)\left(\frac{m}{2+m}\right)^m}{x} - \frac{m(m-2)}{2}. \tag{5.2}$$

It can be easily seen that the above mentioned weight function $Q(x)$ satisfies all the conditions of Theorem 4.1. Therefore, we obtain a new optimal general class of fourth-order methods given by

$$\begin{cases} y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)-pf(x_n)}, \\ x_{n+1} = x_n - \frac{m\left((-2+m)f'(y_n)+\left(\frac{m}{2+m}\right)^m(-mf'(x_n)+2(1+m)pf(x_n))\right)f(x_n)}{2\left(\left(\frac{m}{2+m}\right)^m f'(x_n)-f'(y_n)\right)(f'(x_n)-pf(x_n))}. \end{cases} \tag{5.3}$$

This is a new general class of fourth-order optimal methods having the same scaling factor of functions as that of Jarratt's method and does not fail even $f'(x) = 0$. Therefore, these techniques can be used as an alternative to Jarratt's technique or in the cases where Jarratt's technique is not successful. Furthermore, one can easily get many new methods by choosing the different values of the disposable parameter p.

Particular example of optimal family (5.3)

(i) For $p = 0$, family (5.3) reads as

$$\begin{cases} y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \frac{m\left((-2+m)f'(y_n)-m\left(\frac{m}{2+m}\right)^m f'(x_n)\right)f(x_n)}{2f'(x_n)\left(\left(\frac{m}{2+m}\right)^m f'(x_n)-f'(y_n)\right)}. \end{cases} \tag{5.4}$$

This is a well-known Li et al. method (30) [10].

Case 2. Now, we consider the following weight function

$$Q(x) = m - \frac{3m^2}{2} - \frac{27(1+m)^2\left(\frac{m}{2+m}\right)^{2m}}{\left(-\frac{8(1+m)\left(\frac{m}{2+m}\right)^m}{m}\right)\left(x+\frac{(1+m)\left(\frac{m}{2+m}\right)^m}{m}\right)}. \tag{5.5}$$

It can be easily seen that the above mentioned weight function $Q(x)$ satisfies all the conditions of Theorem 4.1. Therefore, we obtain another new optimal general class of fourth-order methods given by

$$\begin{cases} y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)-pf(x_n)}, \\ x_{n+1} = x_n - \frac{mf(x_n)}{2(f'(x_n)-pf(x_n))}\left(2-3m-\frac{B_1}{B_2}\right). \end{cases} \tag{5.6}$$

where

$$B_1 = 54m\left(\frac{m}{2+m}\right)^{2m}(f'(x_n)-p(1+m)f(x_n))^2,$$

$$B_2 = \left(-mf'(y_n)+\left(\frac{m}{2+m}\right)^m((8+m)f'(x_n)-8p(1+m)f(x_n))\right)\times\left(-mf'(y_n)+\left(\frac{m}{2+m}\right)^m((-1+m)f'(x_n)+p(1+m)f(x_n))\right).$$

This is again a new general class of fourth-order optimal methods and one can easily get many new methods by choosing different values of the disposable parameter p.

Particular example of optimal family (5.6)

(i) For $p = 0$, family (5.6) reads as

$$\begin{cases} y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \frac{mf(x_n)\left(m^2(-2+3m)\{f'(y_n)\}^2 - B_3 f'(x_n)f'(y_n) + B_4\{f'(x_n)\}^2\right)}{2f'(x_n)\left((-1+m)\left(\frac{m}{2+m}\right)^m f'(x_n) - mf'(y_n)\right)\left(mf'(y_n) - \left(\frac{m}{2+m}\right)^m (8+m)f'(x_n)\right)}. \end{cases} \tag{5.7}$$

where

$$B_3 = m\left(\frac{m}{2+m}\right)^m(-14 + 17m + 6m^2),$$
$$B_4 = \left(\frac{m}{2+m}\right)^{2m}(16 + 16m + 19m^2 + 3m^3).$$

This is a new fourth-order optimal multipoint iterative method for multiple roots.

Case 3. Now, we consider the following weight function

$$Q(x) = Ax^2 + Bx + C.$$

Then

$$Q'(x) = 2Ax + B, \quad Q''(x) = 2A, \quad Q'''(x) = 0.$$

According to Theorem 4.1, we should solve the following equations:

$$\begin{cases} A\mu^2 + B\mu + C = m, \\ 2A\mu + B = -\frac{m^3\left(\frac{m}{2+m}\right)^{-m}}{4(1+m)}, \\ 2A = \frac{m^4\left(\frac{m}{2+m}\right)^{-2m}}{4(1+m)^2}, \\ Q'''(\mu) = 0. \end{cases} \tag{5.8}$$

After some simplification, we get the values of A, B and C as follows:

$$\begin{cases} A = \frac{m^4\left(\frac{m}{2+m}\right)^{-2m}}{8(1+m)^2}, \\ B = -\frac{3m^3\left(\frac{m}{2+m}\right)^{-m}}{4(1+m)}, \\ C = m(1+m) \end{cases} \tag{5.9}$$

and thus we obtain the following family of iterative methods:

$$\begin{cases} y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n) - pf(x_n)}, \\ x_{n+1} = x_n - \frac{mf(x_n)\left(\frac{m}{2+m}\right)^{-2m}}{8(f'(x_n) - pf(x_n))(f'(x_n) - (1+m)pf(x_n))^2}\Big[m^3\{f'(y_n)\}^2 - 2m^2\left(\frac{m}{2+m}\right)^m \times (f'(x_n)(3+m) - 3f(x_n)(1+m)p)f'(y_n) \\ \qquad + \left(\frac{m}{2+m}\right)^{2m}(\{f'(x_n)\}^2(8 + 8m + 6m^2 + m^3) - 2f(x_n)f'(x_n)(8 + 16m + 11m^2 + 3m^3)p + 8\{f'(x_n)\}^2(1+m)^3p^2)\Big]. \end{cases} \tag{5.10}$$

This is again a new general class of fourth-order optimal methods and one can easily get many new methods by choosing different values of the disposable parameter p.

Special case of optimal family (5.10)

(i) For $p = 0$, family (5.10) reads as

$$\begin{cases} y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \frac{m\left[\frac{m}{2+m}\right)^{-2m} f(x_n)}{8\{f'(x_n)\}^3}\left(m^3\{f'(y_n)\}^2 - 2m^2\left(\frac{m}{2+m}\right)^m(3+m)f'(x_n)f'(y_n) + \left(\frac{m}{2+m}\right)^{2m}(8 + 8m + 6m^2 + m^3)\{f'(x_n)\}^2\right]. \end{cases} \tag{5.11}$$

This is a well-known Zhou et al. method (11) [13].

Case 4. Since p is a free disposable parameter in scheme (5.1). Therefore, for $p = 0$ in scheme (5.1), we get

$$\begin{cases} y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} Q\left(\frac{(m+1)(f'(y_n)}{f'(x_n)} - (m+1)\left(\frac{m}{m+2}\right)^m\right). \end{cases} \tag{5.12}$$

This is a well-known Zhou et al. family of methods [13].

Remark 1. The first most striking feature of this contribution is that we have developed one point family of order two and multipoint optimal general class of fourth-order methods for the first time which will converge even though the guess is far from root or the derivative is small in the vicinity of the required root.

Remark 2. Here, we should note that one can easily develop several new optimal families of higher-order methods from scheme (5.1) by choosing different type of weight functions, permitting $f'(x) = 0$ in the vicinity of the required root.

Remark 3. Li et al. method and Zhou et al. family of methods (method (5.11)) are obtained as the special cases of our proposed schemes (5.3) and (5.10) respectively.

Remark 4. One should note that all the proposed families require one evaluations of the function and two of it's first-order derivative viz. $f(x_n)$, $f'(x_n)$ and $f'(y_n)$ per iteration. Theorem 4.1 shows that the proposed schemes are optimal with fourth-order convergence, as expected by Kung-Traub conjecture [7]. Therefore, the proposed class of methods has an efficiency index which equals 1.587.

Remark 5. If at any point during the search, $f'(x) = 0$, Newton's method and it's variants would fail due to division by zero. Our methods do not exhibit this type of behaviour.

Remark 6. Further, it is investigated that our proposed scheme (5.1) gives very good approximation to the root when $|p|$ is small. This is because that, for small values of p, slope or angle of inclination of straight line with *x–axis* becomes smaller, i.e. as $p \to 0$, the straight line tends to *x–axis*. This means that our next approximation will move faster towards the desired root. For large values of p, the formula still works but takes more number of iterations as compared to the smaller values of p.

6. Numerical experiments

In this section, we shall check the effectiveness of the new optimal methods. We employ the present methods, namely, family (5.3) and family (5.6) for $|p| = 1$ denoted by, MLM, MM respectively to solve nonlinear equations. We compare them with existing robust methods namely, Rall's method (RM) [1], method (5.11) (ZM1), Zhou et al. method (12) (ZM2) [13], method (2.2) (SM), Li et al. method (69) (LM1) [9] and method (5.4) (LM2) respectively. For better comparisons of our proposed methods, we have given two comparsion tables in each example: one is corresponding to absolute error value of given nonlinear functions (with the same total number of functional evaluations =12) and other is with respect to number of iterations taken by each method to obtain the root correct up to 35 significant digits. All computations have been performed using the programming package *Mathematica* 9 with multiple precision arithmetic.

Example 6.1. Consider the following 6×6 matrix

$$A = \begin{bmatrix} 5 & 8 & 0 & 2 & 6 & -6 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 6 & 18 & -1 & 1 & 13 & -9 \\ 3 & 6 & 0 & 4 & 6 & -6 \\ 4 & 14 & -2 & 0 & 11 & -6 \\ 6 & 18 & -2 & 1 & 13 & -8 \end{bmatrix}.$$

The corresponding characteristic polynomial of this matrix is as follows:

$$f_1(x) = (x-1)^3(x-2)(x-3)(x-4). \tag{6.1}$$

Its characteristic equation has one multiple root at $x = 1$ of multiplicity three. It can be seen that (RM), (ZM1), (ZM2), (SM), (LM1) and (LM2) methods do not necessarily converge to the root that is nearest to the starting value. For example, (LM1) and (LM2) with initial guess $x_0 = 1.6$ diverge while (ZM1), (ZM2), (SM) converge to the root after finite number of iterations. Similarly, (RM), (ZM1), (ZM2), (SM), (LM1) and (LM2) with initial guess $x_0 = 1.7$ are divergent. Our methods do not exhibit this type of behaviour.

$f(x)$	x_0	*RM*	*ZM*1	*ZM*2	*SM*	*LM*1	*LM*2	*MM* $\|p\| = 1$	*MLM* $\|p\| = 1$
Comparison of different iterative methods with the same total number of functional evaluations (TNFE=12)									
$f_1(x)$	0.4	1.48e − 110	5.20e − 353	4.12e − 358	2.02e − 361	2.62e − 365	3.68e − 367	9.37e − 557	4.01e − 550
	0.6	2.75e − 136	3.22e − 446	1.01e − 451	2.84e − 455	2.20e − 459	2.56e − 461	1.09e − 702	1.55e − 696
	1.3	6.11e − 121	6.29e − 307	8.88e − 310	1.67e − 311	2.76e − 313	4.70e − 314	9.53e − 607	5.39e − 594
	1.6	1.04e − 29	6.66e + 12	8.20e + 3	1.34e − 17	CUR	CUR	1. 01e − 2	3.23e − 6
	1.7	D	D	D	D	D	1.06e − 16	2.90e − 25	9.44e − 1
Comparison of different iterative methods with respect to number of iteration									
$f_1(x)$	0.4	6	4	4	4	4	4	3	3
	0.6	6	4	4	4	4	4	3	3
	1.3	6	4	4	4	4	4	3	3
	1.6	8	11	9	6	D	D	7	7
	1.7	D	D	D	D	D	6	6	8

Example 6.2. Consider the following 5×5 matrix

$$B = \begin{bmatrix} 29 & 14 & 2 & 6 & -9 \\ -47 & -22 & -1 & -11 & 13 \\ 19 & 10 & 5 & 4 & -8 \\ -19 & -10 & -3 & -2 & 8 \\ 7 & 4 & 3 & 1 & -3 \end{bmatrix}.$$

The corresponding characteristic polynomial of this matrix is as follows:

$$f_2(x) = (x-2)^4(x+1). \tag{6.2}$$

Its characteristic equation has one multiple root at $x = 2$ of multiplicity four. It can be seen that all the mentioned methods fail with initial guess $x_0 = -0.4$. Our methods do not exhibit this type of behaviour.

$f(x)$	x_0	*RM*	*ZM*1	*ZM*2	*SM*	*LM*1	*LM*2	*MM* $\|p\| = 1$	*MLM* $\|p\| = 1$
Comparison of different iterative methods with the same total number of functional evaluations (TNFE=12)									
$f_2(x)$	−0.4	F	F	F	F	F	F	1.51e − 143	8.46e − 79
	1.0	7.41e − 244	2.628e − 151	1.52e − 151	1.05e − 151	3. 72e − 614	1.58e − 616	6.72e − 653	2.68e − 656
	1.1	5.11e − 259	1.908e − 681	1.57e − 682	2.99e − 683	3. 07e − 684	1.18e − 684	7.20e − 687	2.78e − 690
	2.9	3.49e − 302	1.56e − 929	7.32e − 931	9.23e − 932	4.74e − 933	1.29e − 933	1.06e − 709	1.86e − 674
Comparison of different iterative methods with respect to number of iteration									
$f_2(x)$	−0.4	F	F	F	F	F	F	4	4
	1.0	6	3	3	3	3	3	3	3
	1.1	6	3	3	3	3	3	3	3
	2.9	5	3	3	3	3	3	3	3

Example 6.3. $f_3(x) = \sin^2 x$.

This equation has an infinite number of roots with multiplicity two but our desired root is $r_m = 0$. It can be seen that (RM), (ZM1), (ZM2), (SM), (LM1) and (LM2) methods do not necessarily converge to the root that is nearest to the starting value. For example, (RM), (ZM1), (ZM2), (SM), (LM1) and (LM2) methods with initial guess $x_0 = -1.51$ converge to 15.7079 ...,12069.9989 ...,493.2300 ..., 6.2832 ..., − 3.1416 ..., − 3.1416 ..., respectively, far away from the required root zero. Similarly, (RM), (ZM1), (ZM2), (SM), (LM1) and (LM2) methods with initial guess $x_0 = 1.51$ converge to 12069.9989 ...,493.2300 ..., − 6.2832 ..., 3.1416 ... and 3.1416 ... respectively. Our methods do not exhibit this type of behaviour.

$f(x)$	x_0	*RM*	*ZM*1	*ZM*2	*SM*	*LM*1	*LM*2	*MM* $\lvert p\rvert = 1$	*MLM* $\lvert p\rvert = 1$
Comparison of different iterative methods with the same total number of functional evaluations									
$f_3(x)$	−1.51	CUR	CUR	CUR	CUR	CUR	CUR	1.27e − 318	8.39e − 288
	− 0.6	1.90e − 638	4.33e − 532	2.35e − 536	1.99e − 538	2. 55e − 540	2.55e − 540	1.13e − 342	2.35e − 326
	0.3	1.10e − 1102	4.42e − 957	8.50e − 959	1.19e − 959	1.73e − 960	1.73e − 960	8.19e − 454	1.77e − 433
	1.51	CUR	CUR	CUR	CUR	CUR	CUR	1.27e − 318	8.34e − 288
Comparison of different iterative methods with respect to number of iteration									
$f_3(x)$	−1.51	CUR	CUR	CUR	CUR	CUR	CUR	3	3
	−0.6	4	3	3	3	3	3	3	3
	0.3	4	3	3	3	3	3	3	3
	1.51	CUR	CUR	CUR	CUR	CUR	CUR	3	3

Example 6.4. $f_4(x) = (e^{-x} + \sin x)^3$.

This equation has an infinite number of roots with multiplicity three but our desired root r_m = 3.183063011933363591939186995636394 6. It can be seen that (ZM1), (ZM2), and (SM) methods do not necessarily converge to the root that is nearest to the starting value. For example, (RM), (ZM1), (ZM2) and (SM) methods with initial guess x_0 = 1.70 converge to undesired root 6.2813 . . .,40.8407 . . . , 7875.9727 . . . ,1060394.3347 . . . , while (LM2) converge to the required root after finite number of iteration but (LM1) diverges to the required root. Similarly, (RM), (LM1) and (LM2) methods with initial guess x_0 = 4.40 converges to undesired root 12.2912 . . . , 267.0353 . . . and 9.4248 . . . respectively while (ZM1), (ZM2) and (SM) methods are divergent. Our methods do not exhibit this type of behaviour.

$f(x)$	x_0	*RM*	*ZM*1	*ZM*2	*SM*	*LM*1	*LM*2	*MM* $\lvert p\rvert = 1$	*MLM* $\lvert p\rvert = 1$
Comparison of different iterative methods with the same total number of functional evaluations									
$f_4(x)$	1.70	CUR	CUR	CUR	CUR	D	3.51e + 2	1.44e − 278	1.25e − 278
	2.50	1.61e − 221	3.33e − 444	9.60e − 445	4.62e − 445	2.23e − 445	1.64e − 445	2.56e − 469	3.40e − 466
	3.80	1.27e − 260	7.18e − 424	1.52e − 424	6.15e − 425	2. 51e − 425	1.73e − 425	7.96e − 523	2.94e − 521
	4.40	CUR	D	D	D	CUR	CUR	6.18e − 405	4.60e − 395
Comparison of different iterative methods with respect to number of iteration									
$f_4(x)$	1.70	CUR	CUR	CUR	CUR	D	14	4	4
	2.50	5	3	3	3	3	3	3	3
	3.80	5	4	4	4	4	3	3	3
	4.40	CUR	D	D	D	CUR	CUR	4	4

Example 6.5. $f_5(x) = (5\tan^{-1}x - 4x)^8$.

This equation has an finite number of roots with multiplicity eight but our desired root is r_m = 0.94913461128828951372581521479848875. It can be seen that (RM), (ZM1), (ZM2), (SM), (LM1) and (LM2) methods do not necessarily converge to the root that is nearest to the starting value. For example, all the mentioned methods with initial guess x_0 = 0.5 fail to converge the required root but our methods converge the required root after finite number of iteration.

$f(x)$	x_0	*RM*	*ZM*1	*ZM*2	*SM*	*LM*1	*LM*2	*MM* $\lvert p\rvert = 1$	*MLM* $\lvert p\rvert = 1$
Comparison of different iterative methods with the same total number of functional evaluations									
$f_5(x)$	0.5	F	F	F	F	F	F	1.79e − 11	2.04e − 19
	0.7	2.58e − 238	1.63e − 248	1.63e − 248	1.63e − 248	1. 75e − 248	1.81e − 248	4.43e − 448	1.66e − 447
	1.0	3.59e − 685	1.308e − 2296	2.23e − 2297	5.43e − 2298	5.49e − 2300	1.75e − 2300	3.28e − 2515	6.92e − 2513
	1.2	1.43e − 379	9.26e − 1136	2.04e − 1136	6.05e − 1137	1.09e − 1138	3.98e − 1139	4.11e − 1396	1.01e − 1393
Comparison of different iterative methods with respect to number of iteration									
$f_5(x)$	0.5	F	F	F	F	F	F	7	6
	0.7	7	5	5	5	5	5	4	4
	1.0	5	3	3	3	3	3	3	3
	1.2	6	3	3	3	3	3	3	3

Acknowledgment

The authors are thankful to the referee for his useful technical comments and valuable suggestions, which led to a significant improvement of the paper.

References

[1] L.B. Rall, Convergence of Newton's process to multiple solutions, Numer. Math. 9 (1966) 23–37.
[2] X. Wu, H. Wu, On a class of quadratic convergence iteration formulae without derivatives, Appl. Math. Comput. 107 (2000) 77–80.
[3] Mamta, V. Kanwar, V.K. Kukreja, S. Singh, On a class of quadratically convergent iteration formulae, Appl. Math. Comput. 166 (2005) 633–637.
[4] S. Kumar, V. Kanwar, S.K. Tomar, S. Singh, Geometrically constructed families of Newtons method for unconstrained optimization and nonlinear equations, Int. J. Math. Math. Sci. 2011 (2011) 972537, 9 pages.
[5] A.M. Ostrowski, Solutions of Equations and System of Equations, Academic Press, New York, 1960.
[6] J.F. Traub, Iterative Methods for the Solution of Equations, Prentice-Hall, Englewood Cliffs, NJ, 1964.
[7] H.T. Kung, J.F. Taub, Optimal order of one-point and multipoint iteration, J. ACM 21 (1974) 643–651.
[8] S. Kumar, V. Kanwar, S. Singh, On some modified families of multipoint iterative methods for multiple roots of nonlinear equations, Appl. Math. Comput. 218 (2012) 7382–7394.
[9] S.G. Li, L.Z. Cheng, B. Neeta, Some fourth-order nonlinear solvers with closed formulae for multiple roots, Comput. Math. Appl. 59 (2010) 126–135.
[10] S. Li, X. Liao, L. Cheng, A new fourth-order iterative method for finding multiple roots of nonlinear equations, Appl. Math. Comput. 215 (2009) 1288–1292.
[11] B. Neta, A.N. Jhonson, High order nonlinear solver for multiple roots, Comput. Math. Appl. 55 (2008) 2012–2017.
[12] J.R. Sharma, R. Sharma, Modified Jarratt method for computing multiple roots, Appl. Math. Comput. 217 (2010) 878–881.
[13] X. Zhou, X. Chen, Y. Song, Constructing higher-order methods for obtaining the multiple roots of nonlinear equations, J. Comput. Appl. Math. 235 (2011) 4199–4206.
[14] P. Jarratt, Some efficient fourth-order multipoint methods for solving equations, BIT (1969) 119–124.