

Research Article

Another Simple Way of Deriving Several Iterative Functions to Solve Nonlinear Equations

Ramandeep Behl,¹ V. Kanwar,² and Kapil K. Sharma¹

¹ Department of Mathematics, Panjab University, Chandigarh 160 014, India

² University Institute of Engineering and Technology, Panjab University, Chandigarh 160 014, India

Correspondence should be addressed to V. Kanwar, vmithil@yahoo.co.in

Received 25 June 2012; Revised 20 September 2012; Accepted 4 October 2012

Academic Editor: Alicia Cordero

Copyright © 2012 Ramandeep Behl et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present another simple way of deriving several iterative methods for solving nonlinear equations numerically. The presented approach of deriving these methods is based on exponentially fitted osculating straight line. These methods are the modifications of Newton's method. Also, we obtain well-known methods as special cases, for example, Halley's method, super-Halley method, Ostrowski's square-root method, Chebyshev's method, and so forth. Further, new classes of third-order multipoint iterative methods free from a second-order derivative are derived by semidiscrete modifications of cubically convergent iterative methods. Furthermore, a simple linear combination of two third-order multipoint iterative methods is used for designing new optimal methods of order four.

1. Introduction

Various problems arising in mathematical and engineering science can be formulated in terms of nonlinear equation of the form

$$f(x) = 0. \quad (1.1)$$

To solve (1.1), we can use iterative methods such as Newton's method [1–19] and its variants, namely, Halley's method [1–3, 5, 6, 8, 9], Euler's method (irrational Halley's method) [1, 3], Chebyshev's method [1, 2], super-Halley method [2, 4] as Ostrowski's square-root method [5, 6], and so forth, available in the literature.

Among these iterative methods, Newton's method is probably the best known and most widely used algorithm for solving such problems. It converges quadratically to a simple

root and linearly to a multiple root. Its geometric construction consists in considering the straight line

$$y = ax + b \quad (1.2)$$

and then determining the unknowns a and b by imposing the tangency conditions

$$y(x_n) = f(x_n), \quad y'(x_n) = f'(x_n), \quad (1.3)$$

thereby obtaining the tangent line

$$y(x) - f(x_n) = f'(x_n)(x - x_n), \quad (1.4)$$

to the graph of $f(x)$ at $(x_n, f(x_n))$.

The point of intersection of this tangent line with x -axis gives the celebrated Newton's method

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \quad n \geq 0. \quad (1.5)$$

Newton's method for multiple roots appears in the work of Schröder [19], which is given as

$$x_{n+1} = x_n - \frac{f(x_n)f'(x_n)}{f'^2(x_n) - f(x_n)f''(x_n)}. \quad (1.6)$$

This method has a second-order convergence, including the case of multiple roots. It may be obtained by applying Newton's method to the function $u_f(x) = f(x)/f'(x)$, which has simple roots in each multiple root of $f(x)$. The well-known third-order methods which entail the evaluation of $f''(x)$ are close relatives of Newton's method and can be obtained by admitting geometric derivation [1, 2, 5] from the different quadratic curves, for example, parabola, hyperbola, circle or ellipse, and so forth.

The purpose of the present work is to provide some alternative derivations to the existing third-order methods through an exponentially fitted osculating straight line. Some other new formulas are also presented. Here, we will make use of symbolic computation in the programming package Mathematica 7 to derive the error equations of various iterative methods.

Before starting the development of iterative scheme, we would like to introduce some basic definitions.

Definition 1.1. A sequence of iterations $\{x_n \mid n \geq 0\}$ are said to converge with order $P \geq 1$ to a point r if

$$|r - x_{n+1}| \leq C|r - x_n|^P, \quad n \geq 0, \quad (1.7)$$

for some $C > 0$. If $P = 1$, the sequence is said to converge linearly to r . In that case, we require $C < 1$; the constant C is called the rate of linear convergence of x_n to r .

Definition 1.2. Let $e_n = x_n - r$ be the error in the n th iteration; one calls the relation

$$e_{n+1} = Ce_n^P + O(e_n^{P+1}), \quad (1.8)$$

as the error equation. If we can obtain the error equation for any iterative method, then the value of P is its order of convergence.

Definition 1.3. Let d be the number of new pieces of information required by a method. A piece of information typically is any evaluation of a function or one of its derivatives. The efficiency of the method is measured by the concept of efficiency index [6] and is defined by

$$E = P^{1/d}, \quad (1.9)$$

where P is the order of the method.

2. Development of the Methods

Two equivalent derivations for the third-order iterative methods through exponentially fitted osculating straight line are presented below.

Case I. Consider an exponentially fitted osculating straight line in the following form:

$$y(x) = e^{p(x-x_n)} [A(x - x_n) + B], \quad (2.1)$$

where $p \in \mathbb{R}$, $|p| < \infty$, and A and B are arbitrary constants. These constants will be determined by using the tangency conditions at the point $x = x_n$.

If an exponentially fitted osculating straight line given by (2.1) is tangent to the graph of the equation in question, that is, $f(x) = 0$ at $x = x_n$, then we have

$$y^{(k)}(x_n) = f^{(k)}(x_n), \quad k = 0, 1, 2. \quad (2.2)$$

Therefore, we obtain

$$A = f'(x_n) - pf(x_n), \quad B = f(x_n) \quad (2.3)$$

and a quadratic equation in p as follows:

$$p^2 f(x_n) - 2pf'(x_n) + f''(x_n) = 0. \quad (2.4)$$

Suppose that the straight line (2.1) meets the x -axis at $x = x_{n+1}$, then

$$y(x_{n+1}) = 0, \quad (2.5)$$

and it follows from (2.1) that

$$A(x_{n+1} - x_n) + B = 0. \quad (2.6)$$

From (2.6) and if $A \neq 0$, we get

$$x_{n+1} = x_n - \frac{B}{A}, \quad (2.7)$$

or

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - pf(x_n)}, \quad n \geq 0. \quad (2.8)$$

This is the well-known one-parameter family of Newton's method [2]. This family converges quadratically under the condition $f'(x_n) - pf(x_n) \neq 0$, while $f'(x_n) = 0$ is permitted in some points. For $p = 0$, we obtain Newton's method. The error equation of Scheme (2.8) is given by

$$e_{n+1} = (p - c_2)e_n^2 + O(e_n^3), \quad (2.9)$$

where $e_n = x_n - r$, $c_k = (1/k!)(f^{(k)}(r)/f'(r))$, $k = 2, 3, \dots$, and $x = r$ is the root of nonlinear (1.1). In order to obtain the quadratic convergence, the entity in the denominator should be the largest in magnitude. It is straightforward to see from the above error equation (2.9) that for $p = c_2 = (f''(x_n)/2f'(x_n))$, we obtain the well-known third-order Halley's method.

If we apply the well-known Newton's method (1.5) to the modified function $u_f(x) = f(x_n)/(f'(x_n) - pf(x_n))$, we get another iterative method as

$$x_{n+1} = x_n - \frac{f(x_n)\{f'(x_n) - pf(x_n)\}}{f'^2(x_n) - f(x_n)f''(x_n)}. \quad (2.10)$$

This is a new one-parameter modified family of Schröder's method [19] for an equation having multiple roots of multiplicity $m > 1$ unknown. It is interesting to note that by ignoring the term p , method (2.10) reduces to Schröder's method. It is easy to verify that this method is also an order two method, including the case of multiple zeros. Theorem 2.1 indicates that what choice on the disposable parameter p in family (2.10), the order of convergence will reach at least the second and third order, respectively.

Theorem 2.1. *Let $f : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ have at least three continuous derivatives defined on an open interval I , enclosing a simple zero of $f(x)$ (say $x = r \in I$). Assume that initial guess $x = x_0$ is sufficiently close to r and $f'(x_n) - pf(x_n) \neq 0$ in I . Then an iteration scheme defined by formula (2.10) has at least a second order convergence and will have a third-order convergence when $p = c_2$. It satisfies the following error equation*

$$e_{n+1} = (p - c_2)e_n^2 + 2(c_2^2 - 4c_3)e_n^3 + O(e_n^4), \quad (2.11)$$

where $p \in \mathbb{R}$ is a free disposable parameter, $e_n = x_n - r$, and $c_k = (1/k!)(f^{(k)}(r)/f'(r))$, $k = 2, 3, \dots$

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

$$\begin{aligned} f(x_n) &= f'(r)\left(e_n + c_2e_n^2 + c_3e_n^3\right) + O\left(e_n^4\right), \\ f'(x_n) &= f'(r)\left(1 + 2c_2e_n + 3c_3e_n^2\right) + O\left(e_n^3\right), \end{aligned} \quad (2.12)$$

respectively.

Furthermore, we have

$$f''(x_n) = f'(r)(2c_2 + 6c_3e_n) + O\left(e_n^2\right). \quad (2.13)$$

From (2.12) and (2.13), we have

$$\frac{f(x_n)(f'(x_n) - pf(x_n))}{f'^2(x_n) - f(x_n)f''(x_n)} = e_n - (p - c_2)e_n^2 - (2c_2^2 - 4c_3)e_n^3 + O\left(e_n^4\right). \quad (2.14)$$

Finally, The using above equation (2.14) in our proposed scheme (2.10), we get

$$\begin{aligned} e_{n+1} &= e_n - \frac{f(x_n)(f'(x_n) - pf(x_n))}{f'^2(x_n) - f(x_n)f''(x_n)} \\ &= e_n - \left(e_n - (p - c_2)e_n^2 - (2c_2^2 - 4c_3)e_n^3\right) + O\left(e_n^4\right) \\ &= (p - c_2)e_n^2 + (2c_2^2 - 4c_3)e_n^3 + O\left(e_n^4\right). \end{aligned} \quad (2.15)$$

This reveals that the one-point family of methods (2.10) reaches at least second order of convergence by using only three functional evaluations (i.e., $f(x_n)$, $f'(x_n)$, and $f''(x_n)$) per full iteration. It is straightforward to see that family (2.10) reaches the third order of convergence when $p = c_2$. This completes the proof of Theorem 2.1. \square

Now we wish to construct different third-order iterative methods, which are dependent on the different values of p and are given as below.

Special Cases

(i) When $|p| \ll 1$, then p^2 can be neglected in (2.4), and we get

$$p = \frac{f''(x_n)}{2f'(x_n)}. \quad (2.16)$$

(a) Inserting this value of p in (2.8), we get

$$x_{n+1} = x_n - \frac{2f(x_n)f'(x_n)}{2f'^2(x_n) - f(x_n)f''(x_n)}. \quad (2.17)$$

This is the well-known third-order Halley's method [1–3, 5, 6, 8, 9]. It satisfies the following error equation:

$$e_{n+1} = (c_2^2 - c_3)e_n^3 + O(e_n^4). \quad (2.18)$$

(b) Inserting this value of p in (2.10), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{2f'(x_n)} \left[\frac{2f'^2(x_n) - f(x_n)f''(x_n)}{f'^2(x_n) - f(x_n)f''(x_n)} \right]. \quad (2.19)$$

This is the well-known third-order super-Halley method [1, 2, 4]. It satisfies the following error equation:

$$e_{n+1} = -c_3e_n^3 + O(e_n^4). \quad (2.20)$$

(ii) When we take p as an implicit function, then the values of p are based on the idea of successive approximations. Therefore, we get another value of p from (2.4) as

$$p = \frac{f''(x_n)}{2f'(x_n) - pf(x_n)}. \quad (2.21)$$

Here, the function $f''(x_n)/(2f'(x_n) - pf(x_n))$ which occurs on the right cannot be computed till p is known. To get round off the difficulty, we substitute the value of $p = f''(x_n)/2f'(x_n)$ from a previously obtained value in (2.16). Therefore, the new modified value of p is

$$p = \frac{2f'(x_n)f''(x_n)}{4f'^2(x_n) - f(x_n)f''(x_n)}. \quad (2.22)$$

(a) Inserting this value of p in (2.8), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{4f'^2(x_n) - f(x_n)f''(x_n)}{4f'^2(x_n) - 3f(x_n)f''(x_n)} \right]. \quad (2.23)$$

This is a new third-order iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{c_2^2 - 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.24)$$

(b) Again inserting this value of p in (2.10), we get

$$x_{n+1} = x_n - \frac{f(x_n)f'(x_n) \{4f'^2(x_n) - 3f(x_n)f''(x_n)\}}{\{4f'^2(x_n) - f(x_n)f''(x_n)\} \{f'^2(x_n) - f(x_n)f''(x_n)\}}. \quad (2.25)$$

This is again a new third-order iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{c_2^2 - 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.26)$$

(iii) Similarly, from (2.4), one can get another value of p as

$$p = \frac{p^2 f(x_n) + f''(x_n)}{2f'(x_n)}. \quad (2.27)$$

Again inserting the previously obtained value of $p = f''(x_n)/2f'(x_n)$ in the right-hand side of (2.16), we get another modified value of p as

$$p = \frac{f''(x_n) \{4f'^2(x_n) + f(x_n)f''(x_n)\}}{8f'^3(x_n)}. \quad (2.28)$$

(a) Inserting this value of p in (2.8), we get

$$x_{n+1} = x_n - \frac{8f(x_n)f'^3(x_n)}{8f'^4(x_n) - \{4f'^2(x_n) + f(x_n)f''(x_n)\}f(x_n)f''(x_n)}. \quad (2.29)$$

This is a new third-order iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{c_2^2 - 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.30)$$

(b) Again inserting this value of p in (2.10), we obtain

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{8f'^4(x_n) - f(x_n)f''(x_n)\{4f'^2(x_n) + f(x_n)f''(x_n)\}}{8f'^2(x_n)\{f'^2(x_n) - f(x_n)f''(x_n)\}} \right]. \quad (2.31)$$

This is again a new third-order iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{c_2^2 - 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.32)$$

(iv) Now we solve the quadratic equation (2.4) for general values of p . Hence, we get

$$p = \frac{f'(x_n) \pm \sqrt{f'^2(x_n) - f(x_n)f'(x_n)}}{f(x_n)}. \quad (2.33)$$

(a) Inserting these values of p either in (2.4) or (2.10), we get the well-known third-order Ostrowski's square-root method [5] as

$$x_{n+1} = x_n \pm \frac{f(x_n)}{\sqrt{f'^2(x_n) - f(x_n)f''(x_n)}}. \quad (2.34)$$

It satisfies the following error equation:

$$e_{n+1} = \left(\frac{c_2^2 - 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.35)$$

(v) When we rationalize the numerator of (2.33), we get other values of p as

$$p = \frac{f''(x_n)}{f'(x_n) \mp \sqrt{f'^2(x_n) - f(x_n)f''(x_n)}}. \quad (2.36)$$

Inserting these values of p either in (2.8) or (2.10), we get

$$x_{n+1} = x_n - \frac{f(x_n) \left\{ f'(x_n) \pm \sqrt{f'^2(x_n) - f(x_n)f''(x_n)} \right\}}{f'^2(x_n) \pm f'(x_n) \sqrt{f'^2(x_n) - f(x_n)f''(x_n)} - f(x_n)f''(x_n)}. \quad (2.37)$$

This is a new third-order iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{c_2^2 - 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.38)$$

Case II. In the second case, we have now considered an exponentially fitted osculating straight line in the following form:

$$y(x) = e^{p(y-y(x_n))} [A_1(x - x_n) + A_2], \quad (2.39)$$

where $p \in \mathbb{R}$, $|p| < \infty$, and A_1 and A_2 are arbitrary constants. Adopting the same procedure as above, we get

$$A_1 = f'(x_n) - pf(x_n)f'(x_n), \quad A_2 = f(x_n), \quad (2.40)$$

$$p^2 f(x_n)f'^2(x_n) - p\{f(x_n)f''(x_n) + 2f'^2(x_n)\} + f''(x_n) = 0. \quad (2.41)$$

From (2.40) and (2.41), we get another family of iterative methods given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)(1 - pf(x_n))}. \quad (2.42)$$

This is another new one-parameter family of Newton's method. In order to obtain quadratic convergence, the entity in the denominator should be largest in magnitude. Again note that for $p = 0$, we obtain Newton's method.

Now we apply the well-known Newton's method to the modified function $u_f(x) = f(x_n)/(f'(x_n)\{1 - pf(x_n)\})$, and we will obtain another iterative method as

$$x_{n+1} = x_n - \frac{f(x_n)f'(x_n)\{1 - pf(x_n)\}}{f'^2(x_n) - f(x_n)f''(x_n) + pf^2(x_n)f''(x_n)}. \quad (2.43)$$

This is another new one-parameter modified family of Schröder's method for an equation having multiple roots of multiplicity $m > 1$ unknown. It is interesting to note that by ignoring the term p , (2.43) reduces to Schröder's method. It is easy to verify that this method is also an order two method, including in the case of multiple zeros. Theorems 2.2 and 2.3 indicate that what choice on the disposable parameter p in families (2.42) and (2.43), the order of convergence will reach at least the second order.

Theorem 2.2. Let $f : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ have at least three continuous derivatives defined on an open interval I , enclosing a simple zero of $f(x)$ (say $x = r \in I$). Assume that initial guess $x = x_0$ is sufficiently close to r and $f'(x_n)(1 - pf(x_n)) \neq 0$ in I . Then the family of iterative methods defined by (2.42) has at least a second-order convergence and will have a third-order convergence when $p = f''(x_n)/2$. It satisfies the following error equation:

$$e_{n+1} = (-pf'(r) + c_2)e_n^2 - (p^2f'^2(r) - 2c_2^2 + 2c_3)e_n^3 + O(e_n^4). \quad (2.44)$$

Theorem 2.3. Let $f : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ have at least three continuous derivatives defined on an open interval I , enclosing a simple zero of $f(x)$ (say $x = r \in I$). Assume that initial guess $x = x_0$ is sufficiently close to r and $f'(x_n)(1 - pf(x_n)) \neq 0$ in I . Then the family of iterative methods defined by (2.43) has at least a second-order convergence and will have a third-order convergence when $p = f''(x_n)/2$. It satisfies the following error equation:

$$e_{n+1} = (pf'(r) - c_2)e_n^2 + 2(2pf'(r) + c_2^2 - 2c_3)e_n^3 + O(e_n^4). \quad (2.45)$$

Proof. The proofs of these theorems are similar to the proof of Theorem 2.1. Hence, these are omitted here. \square

Now we wish to construct different third-order iterative methods, which are dependent on the different values of p and are given as follows.

Special Cases

(i) When $|p| \ll 1$, then p^2 can be neglected in (2.41), and we get

$$p = \frac{f''(x_n)}{2f'^2(x_n) + f(x_n)f''(x_n)}. \quad (2.46)$$

(a) Inserting this value of p in (2.42), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}. \quad (2.47)$$

This is the well-known cubically convergent Chebyshev's method [1–4, 9]. It satisfies the following error equation:

$$e_{n+1} = (2c_2^2 - c_3)e_n^3 + O(e_n^4). \quad (2.48)$$

(b) Again inserting this value of p in (2.43), we get

$$x_{n+1} = x_n - \frac{2f(x_n)f'(x_n)}{2f'^2(x_n) - f(x_n)f''(x_n)}. \quad (2.49)$$

This is the already derived well-known cubically convergent Halley's method.

(ii) Adopting the same procedure as in (2.21), we get another value of p from (2.41) as

$$p = \frac{f''(x_n)}{f'^2(x_n)\{2 - pf(x_n)\} + f(x_n)f''(x_n)}. \quad (2.50)$$

Now we substitute $p = f''(x_n)/(2f'^2(x_n) + f(x_n)f''(x_n))$ to get another modified value of p from (2.46) as

$$p = \frac{f''(x_n)\{2f'^2(x_n) + f(x_n)f''(x_n)\}}{4f'^4(x_n) + 3f'^2(x_n)f(x_n)f''(x_n) + f^2(x_n)f''^2(x_n)}. \quad (2.51)$$

(a) Inserting this value of p in (2.43), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{4f'^4(x_n) + 3f(x_n)f'^2(x_n)f''(x_n) + f^2(x_n)f''^2(x_n)}{4f'^4(x_n) + f'^2(x_n)f(x_n)f''(x_n)} \right]. \quad (2.52)$$

This is a new third-order iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{3c_2^2 - 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.53)$$

(b) Again inserting this value of p in (2.43), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{4f'^2(x_n) + f(x_n)f''(x_n)}{4f'^2(x_n) - f(x_n)f''(x_n)} \right]. \quad (2.54)$$

This is again a new third-order iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{3c_2^2 - 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.55)$$

(iii) From (2.41), one can get another value of p as

$$p = \frac{p(f(x_n)f''(x_n) + 2f'^2(x_n)) - f''(x_n)}{pf'^2(x_n)f(x_n)}. \quad (2.56)$$

Inserting this previously obtained value of $p = (f''(x_n)\{2f'^2(x_n) + f(x_n)f''(x_n)\}) / (4f'^4(x_n) + 3f'^2(x_n)f(x_n)f''(x_n) + f^2(x_n)f''^2(x_n))$ in (2.51), we get another modified value of p as

$$p = \frac{f'^2(x_n)f''(x_n) - f'^2(x_n)f(x_n) + f(x_n)f''^2(x_n)}{2f'^4(x_n) + f'^2(x_n)f(x_n)f''(x_n)}. \quad (2.57)$$

(a) Inserting this value of p in (2.42), we get

$$x_{n+1} = x_n - \frac{f(x_n)f'(x_n)\{2f'^2(x_n) + f(x_n)f''(x_n)\}}{2f'^4(x_n) + f'^2(x_n)f^2(x_n) - f^2(x_n)f''^2(x_n)}. \quad (2.58)$$

This is a new third-order iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{1 - 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.59)$$

(b) Again inserting this value of p in (2.43), we get

$$x_{n+1} = x_n - \frac{2f'^5(x_n)f(x_n) + f^3(x_n)f'(x_n)\{f'^2(x_n) - f''^2(x_n)\}}{2f'^6(x_n) - f'^4(x_n)f(x_n)f''(x_n) - f'^2(x_n)f^3(x_n)f''(x_n) + f^3(x_n)f''^3(x_n)}. \quad (2.60)$$

This is again a new third-order iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{6c_2^2 - 1 - 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.61)$$

(iv) Now we solve the quadratic equation (2.41) for the general value of p , and we get other values of p as

$$p = \frac{\{2f'^2(x_n) + f(x_n)f''(x_n)\} \pm \sqrt{\{2f'^2(x_n) + f(x_n)f''(x_n)\}^2 - 4f(x_n)f'^2(x_n)f''(x_n)}}{2f'(x_n)f(x_n)}. \quad (2.62)$$

By inserting the above values of p either in (2.42) or (2.43), we get

$$x_{n+1} = x_n - \frac{2f(x_n)f'(x_n)}{f(x_n)f''(x_n) \pm \sqrt{4f'^4(x_n) + f^2(x_n)f''^2(x_n)}}. \quad (2.63)$$

This is a new third-order iterative method. It satisfies the following error equation:

$$e_{n+1} = -\left(\frac{3c_2^2 + 2c_3}{2} \right) e_n^3 + O(e_n^4). \quad (2.64)$$

3. Third-Order Multipoint Iterative Methods and Their Error Equations

The practical difficulty associated with the above-mentioned cubically convergent methods may be the evaluation of the second-order derivative. Recently, some new variants of Newton's method free from second-order derivative have been developed in [3, 10, 11] and the references cited there in by the discretization of the second-order derivative or by the predictor-corrector approach or by considering different quadrature formulae for the computation of integral arising from Newton's theorem. These multipoint iteration methods calculate new approximations to a zero of a function by sampling $f(x)$ and possibly its derivatives for a number of values of the independent variable at each step.

Here, we also intend to develop new third-order multipoint methods free from the second-order derivative. The main idea of proposed methods lies in the discretization of the second-order derivative involved in the above-mentioned methods.

Expanding the function $f(x_n - f(x_n)/f'(x_n))$ about the point $x = x_n$ by Taylor's expansion, we have

$$f\left(x_n - \frac{f(x_n)}{f'(x_n)}\right) = \frac{f^2(x_n)}{2f'^2(x_n)} f''(x_n) + O\left(\frac{f(x_n)}{f'(x_n)}\right)^3. \quad (3.1)$$

Therefore, we obtain

$$f''(x_n) \approx \frac{2f'^2(x_n)f(y_n)}{f^2(x_n)}, \quad (3.2)$$

where $y_n = x_n - f(x_n)/f'(x_n)$.

Using this approximate value of $f''(x_n)$ into the previously obtained formulae, we get different multipoint iterative methods free from second-order derivative.

Special Cases

- (i) Inserting this approximate value of $f''(x_n)$ (from (3.2)) either in (2.17) or (2.19), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{f(x_n)}{f(x_n) - f(y_n)} \right]. \quad (3.3)$$

This is the well-known third-order Newton-Secant method [3]. It satisfies the following error equation:

$$e_{n+1} = c_2^2 e_n^3 + 3c_2(c_3 - c_2^2)e_n^4 + O(e_n^5). \quad (3.4)$$

- (ii) Inserting this approximate value of $f''(x_n)$ (from (3.2)) in (2.23), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{f(y_n) - 2f(x_n)}{3f(y_n) - 2f(x_n)} \right]. \quad (3.5)$$

This is a new third-order multipoint iterative method having the error equation

$$e_{n+1} = \left(\frac{c_2^2}{2} \right) e_n^3 + \left(\frac{4c_2c_3 - 3c_2^2}{4} \right) e_n^4 + O(e_n^5). \quad (3.6)$$

- (iii) Inserting this approximate value of $f''(x_n)$ (from (3.2)) in (2.25), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{f(x_n)\{3f(y_n) - 2f(x_n)\}}{\{f(y_n) - 2f(x_n)\}\{f(x_n) - 2f(y_n)\}} \right]. \quad (3.7)$$

This is a new third-order multipoint iterative method having the error equation

$$e_{n+1} = \left(\frac{c_2^2}{2} \right) e_n^3 + \left(\frac{4c_2c_3 - 5c_2^3}{4} \right) e_n^4 + O(e_n^5). \quad (3.8)$$

(iv) Inserting this approximate value of $f''(x_n)$ (from (3.2)) in (2.29), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{2f^2(x_n)}{2f^2(x_n) - 2f(y_n)f(x_n) - f^2(y_n)} \right]. \quad (3.9)$$

This is a new third-order multipoint iterative method having the error equation

$$e_{n+1} = \left(\frac{c_2^2}{2} \right) e_n^3 + \left(\frac{2c_2c_3 - c_2^3}{2} \right) e_n^4 + O(e_n^5). \quad (3.10)$$

(v) Inserting this approximate value of $f''(x_n)$ (from (3.2)) in (2.31), we get

$$x_{n+1} = x_n - \frac{2f^2(x_n) - 2f(y_n)f(x_n) - f^2(y_n)}{2f'(x_n)\{f(x_n) - 2f(y_n)\}}. \quad (3.11)$$

This is a new third-order multipoint iterative method having the error equation

$$e_{n+1} = \left(\frac{c_2^2}{2} \right) e_n^3 + \left(\frac{2c_2c_3 - 3c_2^2}{2} \right) e_n^4 + O(e_n^5). \quad (3.12)$$

(vi) Inserting this approximate value of $f''(x_n)$ (from (3.2)) in (2.47), we get

$$x_{n+1} = x_n - \frac{f(x_n) + f(y_n)}{f'(x_n)}. \quad (3.13)$$

This is the well-known third-order Potra-Pták's method [12]. It satisfies the following error equation:

$$e_{n+1} = 2c_2^2e_n^3 + (7c_2c_3 - 9c_2^3)e_n^4 + O(e_n^5). \quad (3.14)$$

(vii) Inserting this approximate value of $f''(x_n)$ (from (3.2)) in (2.52), we get

$$x_{n+1} = x_n - \frac{2f^2(x_n) + 3f(x_n)f(y_n) + 2f^2(y_n)}{f'(x_n)\{2f(x_n) + f(y_n)\}}. \quad (3.15)$$

This is a new third-order multipoint iterative method having the error equation

$$e_{n+1} = \left(\frac{3c_2^2}{2} \right) e_n^3 + \left(\frac{20c_2c_3 - 21c_2^3}{4} \right) e_n^4 + O(e_n^5). \quad (3.16)$$

(viii) Inserting this approximate value of $f''(x_n)$ (from (3.2)) in equation (2.54), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{2f(x_n) + f(y_n)}{2f(x_n) - f(y_n)} \right]. \quad (3.17)$$

This is a new third-order multipoint iterative method having the error equation

$$e_{n+1} = \left(\frac{3c_2^2}{2} \right) e_n^3 + \left(\frac{20c_2c_3 - 23c_2^3}{4} \right) e_n^4 + O(e_n^5). \quad (3.18)$$

(ix) Inserting this approximate value of $f''(x_n)$ (from (3.2)) in (2.58), we get

$$x_{n+1} = x_n - \frac{2f'(x_n)f^2(x_n)\{f(x_n) + f(y_n)\}}{2f'^2(x_n)\{f^2(x_n) - 2f^2(y_n)\} + f^4(x_n)}. \quad (3.19)$$

This is a new third-order multipoint iterative method having the error equation

$$e_{n+1} = \frac{1}{2}e_n^3 + c_2(3c_2^2 - c_3 - 1)e_n^4 + O(e_n^5). \quad (3.20)$$

(x) Inserting this approximate value of $f''(x_n)$ (from (3.2)) in (2.63), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{f(x_n) \left[f^4(x_n) + 2f'^2(x_n) \{ f^2(x_n) - 2f^2(y_n) \} \right]}{2f'^2(x_n) \{ f^3(x_n) - f^2(x_n)f(y_n) + 4f^3(y_n) \} - 2f^4(x_n)f(y_n)} \right]. \quad (3.21)$$

This is a new third-order multipoint iterative method having the error equation

$$e_{n+1} = \left(\frac{6c_2^2 - 1}{2} \right) e_n^3 + 11c_2(c_3 - c_2^2)e_n^4 + O(e_n^5). \quad (3.22)$$

4. Optimal Fourth-Order Multipoint Iterative Methods and Their Error Equations

Now we intend to develop new fourth-order optimal multipoint iterative methods [10, 11, 15–17] for solving nonlinear equations numerically. These multipoint iterative methods are of great practical importance since they overcome theoretical limits of one-point methods concerning the convergence order and computational efficiency. In the case of these multipoint methods, Kung and Traub [13] conjectured that the order of convergence of any multipoint method without memory, consuming n function evaluations per iteration, can not exceed the bound 2^{n-1} (called optimal order). Multipoint methods with this property are called optimal methods. Traub-Ostrowski's method [3], Jarratt's method [14], King's method [11], a family of Traub-Ostrowski's method [10], and so forth are famous optimal fourth

order methods, because they require three function evaluations per step. Traub-Ostrowski's method, Jarratt's method, and King's family are the most efficient fourth-order multipoint iterative methods till date. Nowadays, obtaining new optimal methods of order four is still important, because they have very high efficiency index. For this, we will take linear combination of the Newton-Secant method and the newly developed third-order multipoint iterative methods. Let us denote the Newton-Secant method by N and the methods namely (3.5) to (3.17) by T , respectively, therefore, taking the linear combination of N (the Newton-Secant method) and T (newly developed multipoint methods) as follows:

$$\phi(x_n) = \alpha N + \beta T \quad \text{where } \alpha, \beta \in \mathbb{R}. \quad (4.1)$$

For some particular values of α and β , we get many new fourth-order optimal multipoint iterative methods as follows.

(i) When we take T as a method (3.5) and $(\alpha, \beta) = (-1, 2)$ in (4.1), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{3f(x_n)f(y_n) - 2f^2(y_n) - 2f^2(x_n)}{\{3f(y_n) - 2f(x_n)\}\{f(x_n) - f(y_n)\}} \right]. \quad (4.2)$$

This fourth-order optimal multipoint method is independently derived by Behzad Ghanbari [18]. It satisfies the following error equation

$$e_{n+1} = \left(\frac{3c_2^3 - 2c_2c_3}{2} \right) e_n^4 + O(e_n^5). \quad (4.3)$$

(ii) When we take T as method (3.7) and $(\alpha, \beta) = (-1, 2)$ in (4.1), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{f(x_n)\{4f^2(y_n) + 2f^2(x_n) - 5f(x_n)f(y_n)\}}{\{f(y_n) - 2f(x_n)\}\{f(x_n) - 2f(y_n)\}\{f(y_n) - f(x_n)\}} \right]. \quad (4.4)$$

This is a new fourth-order optimal multipoint iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{c_2^3 - 2c_2c_3}{2} \right) e_n^4 + O(e_n^5). \quad (4.5)$$

Program Code in Mathematica 7 for the Order of Convergence of Method (4.4)

$$\begin{aligned}
 \text{In [1]} : fx &= f1r(e + c_2e^2 + c_3e^3 + c_4e^4), \\
 \text{In [2]} : fdx &= f1r(1 + 2c_2e + 3c_3e^2 + 4c_4e^3 + 5c_5e^4), \\
 \text{In [3]} : u1 &= \text{Series}\left[\frac{fx}{fdx}, \{e, 0, 4\}\right] // \text{Simplify}, \\
 \text{In [4]} : y &= e - u1, \\
 \text{In [5]} : fy &= f1r(y + c_2y^2 + c_3y^3 + c_4y^4), \\
 \text{In [6]} : u2 &= f(x_n) * (4f^2(y_n) + 2f^2(x_n) - 5f(x_n)f(y_n)), \\
 \text{In [7]} : u3 &= (f(y_n) - 2f(x_n))(f(x_n) - 2f(y_n))(f(y_n) - f(x_n)), \\
 \text{In [8]} : u4 &= \text{Series}\left[\frac{u2}{u3}, \{e, 0, 4\}\right], \\
 \text{In [9]} : e1 &= e - u4 // \text{Simplify}, \\
 \text{Out [10]} : &\left(\frac{c_2^3 - 2c_2c_3}{2}\right)e_n^4 + O(e_n^5).
 \end{aligned} \tag{4.6}$$

(iii) When we take T as method (3.9) and $(\alpha, \beta) = (-1, 2)$ in (4.1), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{f(x_n)\{2f^2(x_n) - 2f(x_n)f(y_n) + f^2(y_n)\}}{2f^3(x_n) - 4f^2(x_n)f(y_n) + f(x_n)f^2(y_n) + f^3(y_n)} \right]. \tag{4.7}$$

This is a new fourth-order optimal multipoint iterative method. It satisfies the following error equation:

$$e_{n+1} = (2c_2^3 - c_2c_3)e_n^4 + O(e_n^5). \tag{4.8}$$

(iv) When we take T as method (3.11) and $(\alpha, \beta) = (-1, 2)$ in (4.1), we get

$$x_{n+1} = x_n - \frac{f^3(y_n) + f^2(y_n)f(x_n) - 2f(y_n)f^2(x_n) + f^3(x_n)}{f'(x_n)\{2f^2(y_n) - 3f(y_n)f(x_n) + f^2(x_n)\}}. \tag{4.9}$$

This is a new fourth-order optimal multipoint iterative method. It satisfies the following error equation:

$$e_{n+1} = -c_2c_3e_n^4 + O(e_n^5). \quad (4.10)$$

(v) When we take T as method (3.13) and $(\alpha, \beta) = (-2, 1)$ in (4.1), we get

$$x_{n+1} = x_n - \frac{f^2(x_n) + f^2(y_n)}{f'(x_n)\{f(x_n) - f(y_n)\}}. \quad (4.11)$$

This is a particular case of quadratically convergent King's family [11] of multipoint iterative method for $\gamma = 1$. It satisfies the following error equation:

$$e_{n+1} = (3c_2^3 - c_2c_3)e_n^4 + O(e_n^5). \quad (4.12)$$

(vi) When we take T as a method (3.15) and $(\alpha, \beta) = (3, -2)$ in (4.1), we get

$$x_{n+1} = x_n - \frac{2f^3(x_n) + f^2(x_n)f(y_n) + 2f(x_n)f^2(y_n) + 4f^3(y_n)}{f'(x_n)\{f(x_n) - f(y_n)\}\{2f(x_n) + f(y_n)\}}. \quad (4.13)$$

This is a new fourth-order optimal multipoint iterative method. It satisfies the following error equation:

$$e_{n+1} = (3c_2^3 - c_2c_3)e_n^4 + O(e_n^5). \quad (4.14)$$

(vii) When we take T as a method (3.17) and $(\alpha, \beta) = (3, -2)$ in (4.1), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{2f^2(x_n) - f(x_n)f(y_n) + 2f^2(y_n)}{2f^2(x_n) - 3f(x_n)f(y_n) + f^2(y_n)} \right]. \quad (4.15)$$

This is a new fourth-order optimal multipoint iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{5c_2^3 - 2c_2c_3}{2} \right) e_n^4 + O(e_n^5). \quad (4.16)$$

(viii) Using the approximate value of $f''(x_n)$ (from (3.2)) in (2.19), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{f(x_n) - f(y_n)}{f(x_n) - 2f(y_n)} \right]. \quad (4.17)$$

This is the well-known Traub-Ostrowski's [3] fourth-order optimal multipoint iterative method. It satisfies the following error equation:

$$e_{n+1} = (c_2^3 - c_2c_3)e_n^4 + O(e_n^5). \quad (4.18)$$

Some Other Formulae

In a similar fashion, let us denote Potra-Pták's method by P and the methods, namely, (3.5), (3.9), (3.19) to (3.21), respectively, by T , taking the linear combination of P (Potra-Pták's method) and T (newly developed multipoint methods) as follows:

$$\phi(x_n) = \alpha P + \beta T \quad \text{where } \alpha, \beta \in \mathbb{R}. \quad (4.19)$$

For some particular values of α and β , we get many new other fourth-order optimal multipoint iterative methods as follows.

(ix) When we take T as method (3.9) and $(\alpha, \beta) = (-1, 4)$ in (4.19), we get

$$x_{n+1} = x_n - \frac{6f^3(x_n) + 3f(x_n)f^2(y_n) + f^3(y_n)}{3f'(x_n)\{2f^2(x_n) - 2f(x_n)f(y_n) - f^2(y_n)\}}. \quad (4.20)$$

This is a new fourth-order optimal multipoint iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{7}{3}c_2^3 - c_2c_3\right)e_n^4 + O(e_n^5). \quad (4.21)$$

(x) When we take method (3.5) $\times 3$ - method (3.15), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left[\frac{4f^2(x_n) - 4f(x_n)f(y_n) + 3f^2(y_n)}{4f^2(x_n) - 8f(x_n)f(y_n) + 3f^2(y_n)} \right]. \quad (4.22)$$

This is a new fourth-order optimal multipoint iterative method. It satisfies the following error equation:

$$e_{n+1} = \left(\frac{7}{4}c_2^3 - c_2c_3\right)e_n^4 + O(e_n^5). \quad (4.23)$$

Similarly, we can obtain many other new optimal multipoint fourth-order iterative methods for solving nonlinear equations numerically.

It is straightforward to see that per step these methods require three evaluations of function, namely, two evaluations of $f(x)$ and one of its first-order derivative $f'(x)$. In order to obtain an assessment of the efficiency of our methods, we shall make use of the efficiency index defined by (1.9). For our proposed third-order multipoint iterative methods, we find

Table 1: Test problems.

$f(x)$	$[a, b]$	Root (r)
$f_1(x) = \sin x - \frac{x}{2}$	[1.5, 2]	1.895494222640991
$f_2(x) = \cos x - x$	[0, 2]	0.739085137844086
$f_3(x) = x^3 - 10$	[2, 4]	2.154434680938721
$f_4(x) = 10xe^{-x^2} - 1$	[1, 2]	1.679630637168884
$f_5(x) = (x - 1)^3 - 1$	[1.5, 3.5]	2.000000000000000
$f_6(x) = \tan^{-1}x - x + 1$	[1.5, 3]	2.132267713546753
$f_7(x) = x^4 - x^3 + 11x - 7 = 0$	[0, 1]	0.645023941993713
$f_8(x) = x^5 + 17x = 0$	[-0.5, 1]	0
$f_9(x) = x^3 - \cos x + 2 = 0$	[-2, -1]	-1.172577977180481
$f_{10}(x) = e^{x^2+7x-30} - 1 = 0$	[2.9, 3.5]	3

$P = 3$ and $D = 3$ to get $E = \sqrt[3]{3} \approx 1.442$ which is better than $E = \sqrt{2} \approx 1.414$, the efficiency index of the Newton's method. For the quadratically convergent multipoint iterative methods, we find $P = 4$ and $D = 3$ to get $E = \sqrt[3]{4} \approx 1.587$ which is better than those of most of the third-order methods $E \approx 1.442$ and Newton's method $E \approx 1.414$.

5. Numerical Experiments

In this section, we shall check the effectiveness of the new optimal methods. We employ the present methods, namely, (4.2), (4.9), method (4.15), (4.22) respectively, to solve the following nonlinear equations given in Table 1. We compare them with the methods, namely, Newton's method (NM), Traub-Ostrowski's method (also known as Ostrowski's method) (4.17) (TOM), Jarratt's method (JM), and King's method (KM) for $\gamma = 1/2$ and $\gamma = 1$ respectively. We have also shown the comparison of all methods mentioned above in Table 2. Computations have been performed using C++ in double-precision arithmetic. We use $\varepsilon = 10^{-15}$ as a tolerable error. The following stopping criteria are used for computer programs:

$$(i) |x_{n+1} - x_n| < \varepsilon, (ii) |f(x_{n+1})| < \varepsilon.$$

6. Conclusion

In this paper, we have presented another simple and elegant way of deriving different iterative functions to solve nonlinear equations numerically. This study represents several formulae of third and fourth order and has a well-known geometric derivation. Multipoint iterative methods belong to the class of the most powerful methods since they overcome theoretical limits of one-point methods concerning the convergence order and computational efficiency. The most important class of multipoint methods are optimal methods, which attain the convergence order 2^{n-1} using n function evaluations per iteration. Therefore, fourth-order multipoint iterative methods are the main findings of the present paper in terms of speed and efficiency index. According to Kung-Traub conjecture, these different methods presented in this paper have the maximal efficiency index because only three function values are needed per step. The numerical results presented in Table 2, overwhelmingly, support that these different methods are equally competent to Traub-Ostrowski's method, Jarratt's method, and

Table 2: Total number of iterations to approximate the zero of a function and the total number of function evaluations for various multipoint iterative methods.

Problem	Initial guess	NM	TOM	JM	KM ($\gamma = 1/2$)	KM ($\gamma = 1$)	Method (4.4)	Method (4.9)	Method (4.15)	Method (4.22)
$f_1(x)$	1.5	(4, 8)	(2, 6)	(3, 9)	(3, 9)	(4, 12)	(2, 6)	(2, 6)	(3, 9)	(3, 9)
	2	(3, 6)	(2, 6)	(2, 6)	(2, 6)	(3, 9)	(2, 6)	(2, 6)	(4, 12)	(2, 6)
$f_2(x)$	0	(4, 8)	(2, 6)	(3, 9)	(3, 9)	(3, 9)	(2, 6)	(2, 6)	(3, 9)	(3, 9)
	2	(3, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)
$f_3(x)$	2	(3, 6)	(2, 6)	(3, 9)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)
	4	(5, 10)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)
$f_4(x)$	1	(4, 8)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)
	2	(5, 10)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(2, 6)	(3, 9)	(3, 9)	(3, 9)
$f_5(x)$	1.5	(6, 12)	(3, 9)	(3, 9)	(4, 12)	(5, 15)	(3, 9)	(4, 12)	(4, 12)	(4, 12)
	3.5	(6, 12)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(4, 12)
$f_6(x)$	1.5	(3, 6)	(2, 6)	(2, 6)	(2, 6)	(3, 9)	(2, 6)	(2, 6)	(2, 6)	(3, 9)
	3	(3, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)
$f_7(x)$	0	(3, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)
	1	(3, 6)	(3, 9)	(2, 6)	(3, 9)	(3, 9)	(2, 6)	(2, 6)	(3, 9)	(3, 9)
$f_8(x)$	-0.5	(2, 4)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)
	1	(3, 6)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)
$f_9(x)$	-2	(5, 10)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(3, 9)	(4, 12)	(4, 12)	(3, 9)
	-1	(4, 8)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(2, 6)	(3, 9)	(2, 6)	(2, 6)
$f_{10}(x)$	2.9	(6, 12)	(3, 9)	(6, 18)	(5, 15)	(8, 24)	(3, 9)	(3, 9)	(3, 9)	(3, 9)
	3.5	(11, 22)	(5, 15)	(11, 33)	(6, 18)	(6, 18)	(5, 15)	(6, 18)	(5, 15)	(6, 18)

King's family. By using the same idea, one can obtain other iterative processes by considering different exponentially fitted osculating curves.

Acknowledgments

The authors would like to record their sincerest thanks to the Editor, Professor Alicia Cordero, and anonymous reviewers for their constructive suggestion and remarks which have considerably contributed to the readability of this paper. R. Behl acknowledges further the financial support of CSIR, New Delhi, India.

References

- [1] S. Amat, S. Busquier, and J. M. Gutiérrez, "Geometric constructions of iterative functions to solve nonlinear equations," *Journal of Computational and Applied Mathematics*, vol. 157, no. 1, pp. 197–205, 2003.
- [2] V. Kanwar, S. Singh, and S. Bakshi, "Simple geometric constructions of quadratically and cubically convergent iterative functions to solve nonlinear equations," *Numerical Algorithms*, vol. 47, no. 1, pp. 95–107, 2008.
- [3] J. F. Traub, *Iterative Methods for the Solution of Equations*, Prentice-Hall Series in Automatic Computation, Prentice-Hall, Englewood Cliffs, NJ, USA, 1964.
- [4] J. M. Gutiérrez and M. A. Hernández, "An acceleration of Newton's method: super-Halley method," *Applied Mathematics and Computation*, vol. 117, no. 2-3, pp. 223–239, 2001.

- [5] E. Hansen and M. Patrick, "A family of root finding methods," *Numerische Mathematik*, vol. 27, no. 3, pp. 257–269, 1976/77.
- [6] A. M. Ostrowski, *Solutions of Equations and System of Equations*, Academic Press, New York, NY, USA, 1960.
- [7] W. Werner, "Some improvements of classical iterative methods for the solution of nonlinear equations," in *Numerical Solution of Nonlinear Equations*, vol. 878 of *Lecture Notes in Mathematics*, pp. 426–440, Springer, Berlin, Germany, 1981.
- [8] A. M. Ostrowski, *Solution of Equations in Euclidean and Banach Spaces*, Academic Press, London, UK, 1973.
- [9] L. W. Johnson and R. D. Riess, *Numerical Analysis*, Addison-Wesely, Reading, Mass, USA, 1977.
- [10] V. Kanwar, R. Behl, and K. K. Sharma, "Simply constructed family of a Ostrowski's method with optimal order of convergence," *Computers & Mathematics with Applications*, vol. 62, no. 11, pp. 4021–4027, 2011.
- [11] R. F. King, "A family of fourth order methods for nonlinear equations," *SIAM Journal on Numerical Analysis*, vol. 10, pp. 876–879, 1973.
- [12] F. A. Potra and V. Pták, "Nondiscrete introduction and iterative processes," in *Research Notes in Mathematics*, Pitman, Boston, Mass, USA, 1984.
- [13] H. T. Kung and J. F. Traub, "Optimal order of one-point and multipoint iteration," *Journal of the Association for Computing Machinery*, vol. 21, pp. 643–651, 1974.
- [14] P. Jarratt, "Some fourth-order multipoint methods for solving equations," *BIT*, vol. 9, pp. 434–437, 1965.
- [15] F. Soleymani, "Optimal fourth-order iterative methods free from derivatives," *Miskolc Mathematical Notes*, vol. 12, no. 2, pp. 255–264, 2011.
- [16] M. Sharifi, D. K. R. Babajee, and F. Soleymani, "Finding the solution of nonlinear equations by a class of optimal methods," *Computers & Mathematics with Applications*, vol. 63, no. 4, pp. 764–774, 2012.
- [17] F. Soleymani, S. K. Khattri, and S. Karimi Vanani, "Two new classes of optimal Jarratt-type fourth-order methods," *Applied Mathematics Letters*, vol. 25, no. 5, pp. 847–853, 2012.
- [18] Behzad Ghanbari, "A new general fourth-order family of methods for finding simple roots of nonlinear equations," *Journal of King Saud University-Science*, vol. 23, pp. 395–398, 2011.
- [19] E. Schröder, "Über unendlichviele algorithm zur au osung der gleichungen," *Annals of Mathematics*, vol. 2, pp. 317–365, 1870.