American Journal of Applied Mathematics and Statistics, 2017, Vol. 5, No. 1, 22-32
Available online at http://pubs.sciepub.com/ajams/5/1/5
©Science and Education Publishing
DOI:10.12691/ajams-5-1-5

A Family of Combined Iterative Methods for Solving Nonlinear Equations

Shuping Chen[1,*], Youhua Qian[2]

[1]College of Mathematics, Xiamen University of Technology, xiamen 361024, P. R. China
[2]College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, Jinhua, 321004, P. R. China
*Corresponding author: shupingchen@126.com

Abstract In this article we construct some higher-order modifications of Newton's method for solving nonlinear equations, which is based on the undetermined coefficients. This construction can be applied to any iteration formula. It can be found that per iteration the resulting methods add only one additional function evaluation, their order of convergence can be increased by two or three units. Higher order convergence of our methods is proved and corresponding asymptotic error constants are expressed. Numerical examples, obtained using Matlab with high precision arithmetic, are shown to demonstrate the convergence and efficiency of the combined iterative methods. It is found that the combined iterative methods produce very good results on tested examples, compared to the results produced by the existing higher order schemes in the related literature.

Keywords: *Newton's method, combined iterative methods, nonlinear equations, order of convergence, computational efficiency*

Cite This Article: Shuping Chen, and Youhua Qian, "A Family of Combined Iterative Methods for Solving Nonlinear Equations." *American Journal of Applied Mathematics and Statistics*, vol. 5, no. 1 (2017): 22-32. doi: 10.12691/ajams-5-1-5.

1. Introduction

A variety of problems in different fields of science and technology require to find the solution of nonlinear equations. Iterative methods such as Newton's method are the most used technique. In this paper, we consider a new family of combined iterative methods to find a simple root ξ of a nonlinear equation $f(x)=0$, where f is a real function $f: I \subseteq R \to R$, defined in an open interval I.

The well-known numerical method for the calculation of ξ is the classical Newton's method as given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, n = 0, 1, \cdots, \tag{1}$$

where x_0 is an initial approximation that is sufficiently close to ξ. The convergence order of the classical Newton's method is quadratic for simple roots and linear for multiple roots.

Recently, a number of authors including [1-6] have derived new variants of Newton's methods that offer higher order convergence. These methods are frequently composed of more than two formulas and derived in different ways.

In [6] some fifth order modifications of Newton's method which is extending a general form of third order method are considered. In a similar way, some sixth-order class of modified Ostrowski's methods [7] that improves the order of convergence of Ostrowski's method are presented as follows

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)}, \\ z_n = y_n - \dfrac{f(x_n)}{f(x_n) - 2f(y_n)} \dfrac{f(y_n)}{f'(x_n)}, \\ x_{n+1} = z_n - H(v_n) \dfrac{f(z_n)}{f'(x_n)}, \end{cases} \tag{2}$$

where $v_n = \dfrac{f(y_n)}{f(x_n)}$, and $H(t)$ is a real-valued function for $H(0)=1$, $H'(0)=2$ and $H''(0)<\infty$.

Described below in equation (3) is the three-point sixth-order method [8], which requires only two derivatives and two functions:

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)}, \\ z_n = x_n - \dfrac{2f(x_n)}{f'(x_n) + f'(y_n)}, \\ x_{n+1} = z_n - \dfrac{f'(x_n) + f'(y_n)}{3f'(y_n) - f'(x_n)} \cdot \dfrac{f(z_n)}{f'(x_n)}. \end{cases} \tag{3}$$

This is a three-step method. The first and the second equations of equation (3) compose a third-order method developed by the authors in [9].

It is worth noting that equations (2) and (3) use the technique that consists in applying of new function on the existing iterative schemes. Motivated by the activities in this direction, in this paper, our special attention is paid to the development of a general class of higher-order combined iterative methods. The important feature of our methods is only to add the evaluation of the function at another point. However, their convergence order can be improved and increased above the original level.

This paper is organized as follows. In section 2, the new methods are formulated, and the local convergence theorem is established. Some concrete iterative methods are discussed in section 3. In section 4, the new methods are verified through a number of numerical examples, comparisons of results are also reported to show the effectiveness of the present approach. Finally, the paper ends with conclusions in section 5.

2. The Methods and Analysis of Convergence

Let $\phi(x; f(x), f'(x), f'(y))$ be a function from $R \to R$ with the information f, f' at x and y, this means that the functions $f(x)$, $f'(x)$ and $f'(y)$ at each iteration step are required to evaluate in the computation of ϕ. Now we consider the modification of Newton's method as given below

$$\begin{cases} z_n = \phi(x_n; f(x_n), f'(x_n), f'(y_n)), \\ x_{n+1} = z_n - \dfrac{f(z_n)}{f'(z_n)}, \end{cases} \tag{4}$$

where $z_n = \phi(x; f(x), f'(x), f'(y))$ represents any iterative method whose order of convergence is at least m.

In recent years, a type of multipoint iterative methods have been proposed [9-11]. These methods can be viewed as obtained by approximating $f'(z_n)$ with the expressions $a_1(x_n), \cdots, a_k(x_n)$, where the function $f'(z_n)$ is defined as

$$f'(z_n) = \phi(x_n; a_1(x_n), \cdots, a_k(x_n)). \tag{5}$$

Analogously, in order to derive the new methods, we consider the expression

$$\begin{aligned} f'(z_n) &= Af'(x_n) + Bf'(y_n) \\ &\quad + Cf(z_n) + Df'\left(\frac{x_n + y_n}{2}\right), \end{aligned} \tag{6}$$

for application of the method of undetermined coefficients.

Expand the terms $f'(z_n)$, $f'(y_n)$, $f'\left(\frac{x_n + y_n}{2}\right)$ and $f(z_n)$ about the point x_n up to the third derivatives and collect terms. Upon comparing the coefficients of the derivatives of f at x_n, we get

$$A + B + \alpha C + D = 1, \tag{7}$$

$$C = 0, \tag{8}$$

$$B\beta + \frac{C\alpha^2}{2} + \frac{1}{2}D\beta = \alpha, \tag{9}$$

$$\frac{B\beta^2}{2} + \frac{C\alpha^3}{6} + \frac{1}{8}D\beta^2 = \frac{\alpha^2}{2}, \tag{10}$$

where $\alpha = z_n - x_n$ and $\beta = y_n - x_n$.

Solving equations (7)-(10), we have

$$A = \frac{\beta^2 - 3\alpha\beta + 2\alpha^2}{\beta^2}, \tag{11}$$

$$B = \frac{-\alpha\beta + 2\alpha^2}{\beta^2}, \tag{12}$$

$$C = 0, \tag{13}$$

$$D = \frac{4(\alpha\beta - \alpha^2)}{\beta^2}. \tag{14}$$

Substituting equations (11)-(14) into equation (6), we obtain

$$\begin{aligned} f'(z_n) &= \frac{(\beta^2 - 3\alpha\beta + 2\alpha^2)}{\beta^2} f'(x_n) \\ &\quad - \frac{(\alpha\beta - 2\alpha^2)}{\beta^2} f'(y_n) \\ &\quad + \frac{4(\alpha\beta - \alpha^2)}{\beta^2} f'\left(\frac{x_n + y_n}{2}\right). \end{aligned} \tag{15}$$

By using the arithmetic mean of $f'(x_n)$ and $f'(y_n)$ instead of the midpoint value $f'\left(\frac{x_n + y_n}{2}\right)$ in equation (15), we have

$$f'(z_n) = \frac{(\beta^2 - \alpha\beta)}{\beta^2} f'(x_n) + \frac{\alpha\beta}{\beta^2} f'(y_n). \tag{16}$$

Substituting equation (16) into the second step of equation (4), we propose a higher-order family of combined iterative methods in the following form:

$$\begin{cases} z_n = \phi(x; f(x), f'(x), f'(y)), \\ x_{n+1} = z_n - \dfrac{\beta^2 f(z_n)}{(\beta^2 - \alpha\beta) f'(x_n) + \alpha\beta f'(y_n)}. \end{cases} \tag{17}$$

For this family of methods, the following result can be established.

Theorem 2.1. Let us suppose that $f(x)$ is a sufficiently differentiable function and $f(x)$ has a simple zero ξ. If the initial guess x_0 is close enough to ξ and iteration function $z_n = \phi\left(x_n; f(x_n), f'(x_n), f'(y_n)\right)$ satisfies condition

$$z_n - \xi = Ae_n^m + O\left(e_n^{m+1}\right). \quad (18)$$

If $m < 3$, then the sequence $\{x_k\}$ generated by equation (17) is of order at least $2m$.

If $m \geq 3$, then the sequence $\{x_k\}$ generated by equation (17) is of order at least $m+3$.

Proof. Let $e_n = x_n - \xi$ and $d_n = z_n - \xi$. Using the Taylor expansion and taking into account $f(\xi) = 0$, we arrive at

$$f(x_n) = f'(\xi)\left[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + c_5e_n^5 + c_6e_n^6 + c_7e_n^7 + O\left(e_n^8\right)\right], \quad (19)$$

$$f'(x_n) = f'(\xi)\left[1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + 5c_5e_n^4 + 6c_6e_n^5 + 7c_7e_n^6 + O\left(e_n^7\right)\right], \quad (20)$$

where $c_k = \dfrac{f^k(\alpha)}{k!\,f'(\alpha)}$.

By simple calculations, we have

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\ &= \xi + c_2e_n^2 + 2\left(c_3 - c_2^2\right)e_n^3 \\ &\quad + \left(4c_2^3 - 7c_2c_3 + 3c_4\right)e_n^4 \\ &\quad + \left(-8c_2^4 + 20c_2^2c_3 - 6c_3^2 - 10c_2c_4 + 4c_5\right)e_n^5 \\ &\quad + \left(16c_2^5 - 52c_2^3c_3 + 28c_4c_2^2 - 13c_5c_2 + 33c_3^2c_2 \right. \\ &\quad \left. + 5c_6 - 17c_3c_4\right)e_n^6 + \left(-32c_2^6 + 128c_2^4c_3 - 72c_2^3c_4 \right. \\ &\quad - 126c_3^2c_2^2 + 36c_5c_2^2 + 92c_2c_3c_4 \\ &\quad + 6c_7 - 16c_2c_6 - 12c_4^2 \\ &\quad \left. + 27c_3^3 + 5c_3c_5 - 9c_3^3 - 27c_3c_5\right)e_n^7 + O\left(e_n^8\right), \end{aligned} \quad (21)$$

$$\begin{aligned} f'(y_n) = f'(\xi)\Big[&1 + 2c_2^2e_n^2 - 4c_2\left(c_2^2 - c_3\right)e_n^3 \\ &- \left(11c_2^2c_3 - 8c_2^4 - 6c_2c_4\right)e_n^4 \\ &- 4c_2\left(4c_2^4 - 7c_2^2c_3 + 5c_2c_4 - 2c_5\right)e_n^5 \\ &+ 2c_2\left(16c_2^5 - 40c_2^3c_3 + 30c_2^2c_4 - 13c_2c_5 \right. \\ &\left. + 12c_2c_3^2 + 5c_6 - 8c_3c_4\right)e_n^6 + O\left(e_n^7\right)\Big], \end{aligned} \quad (22)$$

$$f(z_n) = f'(\xi)\left[d_n + c_2d_n^2 + O\left(d_n^3\right)\right], \quad (23)$$

$$\alpha = z_n - x_n = -e_n + (z_n - \xi), \quad (24)$$

$$\begin{aligned} \beta &= y_n - x_n \\ &= -e_n + c_2e_n^2 + 2\left(c_3 - c_2^2\right)e_n^3 \\ &\quad + \left(4c_2^3 - 7c_2c_3 + 3c_4\right)e_n^4 \\ &\quad + \left(-8c_2^4 + 20c_2^2c_3 - 6c_3^2 - 10c_2c_4 + 4c_5\right)e_n^5 \\ &\quad + \left(16c_2^5 - 52c_2^3c_3 + 28c_4c_2^2 - 13c_5c_2 \right. \\ &\quad \left. + 33c_3^2c_2 + 5c_6 - 17c_3c_4\right)e_n^6 \\ &\quad + \left(-32c_2^6 + 128c_2^4c_3 - 72c_2^3c_4 \right. \\ &\quad - 126c_3^2c_2^2 + 36c_5c_2^2 + 92c_2c_3c_4 \\ &\quad - 16c_2c_6 - 12c_4^2 + 6c_7 \\ &\quad \left. + 27c_3^3 + 5c_3c_5 - 9c_3^3 - 27c_3c_5\right)e_n^7 + O\left(e_n^8\right). \end{aligned} \quad (25)$$

Hence, we obtain

$$\begin{aligned} \beta^2 &= e_n^2 - 2c_2e_n^3 + \left(5c_2^2 - 4c_3\right)e_n^4 \\ &\quad + 6\left(3c_2c_3 - 2c_2^3 - c_4\right)e_n^5 \\ &\quad + \left(28c_2^4 - 62c_2^2c_3 + 26c_2c_4 + 16c_3^2 - 8c_5\right)e_n^6 \\ &\quad + \left(-64c_2^5 + 188c_2^3c_3 - 88c_4c_2^2 + 34c_2c_5 \right. \\ &\quad \left. - 106c_2c_3^2 - 10c_6 + 46c_3c_4\right)e_n^7 \\ &\quad + \left(144c_2^6 - 528c_2^4c_3 + 264c_2^3c_4 + 471c_2^2c_3^2 \right. \\ &\quad - 114c_2^2c_5 - 300c_2c_3c_4 + 42c_2c_6 + 33c_4^2 \\ &\quad \left. - 12c_7 - 60c_3^3 + 60c_3c_5\right)e_n^8 + O\left(e_n^9\right), \end{aligned} \quad (26)$$

$$\begin{aligned} \alpha\beta = e_n\Big[&(1 + c_2d_n)e_n - \left(c_2 - 2\left(c_3 - c_2^2\right)d_n\right)e_n^2 \\ &- \left(2\left(c_3 - c_2^2\right) - \left(4c_2^3 - 7c_2c_3 + 3c_4\right)d_n\right)e_n^3 \\ &- \left(4c_2^3 - 7c_2c_3 + 3c_4\right)e_n^4 \\ &- \left(8c_2^4 - 20c_2^2c_3 + 6c_3^2 + 10c_2c_4 - 4c_5\right)d_ne_n^4 \\ &+ \left(8c_2^4 - 20c_2^2c_3 + 6c_3^2 + 10c_2c_4 - 4c_5\right)e_n^5 \\ &+ \left(16c_2^5 - 52c_2^3c_3 + 28c_2^2c_4 - 13c_2c_5 \right. \\ &\left. + 33c_2c_3^2 + 5c_6 - 17c_3c_4\right)d_ne_n^5 \\ &- \left(16c_2^5 - 52c_2^3c_3 + 28c_2^2c_4 \right. \\ &\left. + 33c_2c_3^2 + 5c_6 - 17c_3c_4\right)e_n^6 \\ &+ \left(32c_2^6 - 128c_2^4c_3 + 126c_2^2c_3^2 - 36c_2^2c_5 + 16c_2c_6 \right. \\ &+ 12c_4^2 - 6c_7 - 13c_2c_5 + 72c_2^3c_4 - 92c_2c_3c_4 \\ &\left. - 18c_3^3 + 22c_3c_5\right)e_n^7 - d_n\Big] + O\left(e_n^9\right). \end{aligned} \quad (27)$$

We then have

$$\beta^2-\alpha\beta=e_nd_n-c_2d_ne_n^2-\left(c_2+2\left(c_3-c_2^2\right)d_n\right)e_n^3$$
$$+\left(\left(3c_2^2-2c_3\right)-\left(4c_2^3-7c_2c_3+3c_4\right)d_n\right)e_n^4$$
$$+\left(\left(11c_2c_3-8c_2^3-3c_4\right)\right.$$
$$\left.+\left(8c_2^4-4c_5-20c_2^2c_3+6c_3^2+10c_2c_4\right)d_n\right)e_n^5$$
$$+\left(20c_2^4-4c_5-42c_2^2c_3+16c_2c_4+10c_3^2\right)e_n^6 \quad (28)$$
$$-\left(48c_2^5-136c_2^3c_3+60c_2^2c_4-21c_2c_5+73c_2c_3^2\right.$$
$$\left.+5c_6-29c_3c_4\right)e_n^7+\left(112c_2^6-400c_2^4c_3+192c_2^3c_4\right.$$
$$+345c_2^2c_3^2-78c_2^2c_5-208c_2c_3c_4+26c_2c_6+21c_4^2$$
$$\left.-6c_7-42c_3^3+38c_3c_5\right)e_n^8+O\left(e_n^9\right).$$

Thus, from equations (20), (22), (27) and (28), we attain

$$\left(\beta^2-\alpha\beta\right)f'(x_n)=f'(\xi)\left[e_nd_n+c_2d_ne_n^2\right.$$
$$-(c_2-c_3d_n)e_n^3+\left(\left(c_2^2-2c_3\right)+c_4d_n\right)e_n^4$$
$$+\left(\left(4c_2c_3-2c_2^3-3c_4\right)+c_5d_n\right)e_n^5$$
$$+\left(4c_2^4-11c_2^2c_3+6c_2c_4+4c_3^2-4c_5\right)e_n^6$$
$$-\left(8c_2^5+5c_6-28c_2^3c_3+16c_2^2c_4-8c_2c_5\right. \quad (29)$$
$$\left.+20c_2c_3^2-12c_3c_4\right)e_n^7$$
$$+\left(16c_2^6-68c_2^4c_3+40c_2^3c_4+73c_2^2c_3^2-21c_2^2c_5\right.$$
$$-6c_7-12c_3^3-58c_2c_3c_4+10c_2c_6+9c_4^2$$
$$\left.\left.+16c_3c_5\right)e_n^8+O\left(e_n^9\right)\right],$$

$$\alpha\beta f'(y_n)=f'(\xi)\left[-d_ne_n+(1+c_2d_n)e_n^2\right.$$
$$-\left(c_2-2\left(c_3+2c_2^2\right)d_n\right)e_n^3$$
$$+\left(\left(4c_2^2-2c_3\right)+\left(10c_2^3-11c_2c_3+3c_4\right)d_n\right)e_n^4$$
$$+\left(\left(11c_2c_3-10c_2^3-3c_4\right)\right.$$
$$\left.-\left(12c_2^2c_3-6c_3^2-10c_2c_4+4c_5\right)d_n\right)e_n^5$$
$$+\left(24c_2^4-39c_2^2c_3+6c_3^2+16c_2c_4-4c_5\right)e_n^6 \quad (30)$$
$$-\left(56c_2^5-121c_2^3c_3+60c_2^2c_4\right.$$
$$\left.-21c_2c_5+41c_2c_3^2+5c_6-17c_3c_4\right)e_n^7$$
$$+\left(128c_2^6-358c_2^4c_3+196c_2^3c_4+212c_2^2c_3^2\right.$$
$$-78c_2^2c_5-6c_7-132c_2c_3c_4+26c_2c_6$$
$$\left.\left.+12c_4^2-18c_3^3+22c_3c_5\right)e_n^8+O\left(e_n^9\right)\right].$$

Using equations (29) and (30), we have

$$\left(\beta^2-\alpha\beta\right)f'(x_n)+\alpha\beta f'(y_n)$$
$$=f'(\xi)\left[e_n^2-2c_2e_n^3+\left(5c_2^2-4c_3\right)e_n^4+2c_2d_ne_n^2\right.$$
$$-\left(6c_4+12c_2^3-15c_2c_3\right)e_n^5$$
$$+\left(28c_2^4-50c_2^2c_3+10c_3^2+22c_2c_4-8c_5\right)e_n^6$$
$$+\left(3c_3-4c_2^2\right)d_ne_n^3+\left(10c_2^3-11c_2c_3+4c_4\right)d_ne_n^4$$
$$-\left(12c_2^2c_3-6c_3^2-10c_2c_4+3c_5\right)d_ne_n^5 \quad (31)$$
$$-\left(10c_6-29c_3c_4+64c_2^5-149c_2^3c_3\right.$$
$$\left.+76c_2^2c_4-29c_2c_5+61c_2c_3^2\right)e_n^7$$
$$+\left(144c_2^6-426c_2^4c_3+236c_2^3c_4+285c_2^2c_3^2-99c_2^2c_5\right.$$
$$-190c_2c_3c_4+36c_2c_6+21c_4^2$$
$$\left.\left.-12c_7-30c_3^3+38c_3c_5\right)e_n^8+O\left(e_n^9\right)\right].$$

In the same way, we attain

$$\beta^2f(z_n)$$
$$=f'(\xi)e_n^2d_n\left[1-2c_2e_n+\left(5c_2^2-4c_3\right)e_n^2\right.$$
$$+6\left(3c_2c_3-2c_2^3-c_4\right)e_n^3$$
$$+\left(28c_2^4-62c_2^2c_3+26c_2c_4+16c_3^2-8c_5\right)e_n^4$$
$$+\left(188c_2^3c_3-64c_2^5-88c_4c_2^2+34c_2c_5\right.$$
$$\left.-106c_2c_3^2-10c_6+46c_3c_4\right)e_n^5 \quad (32)$$
$$+c_2d_n-2c_2^2e_nd_n+\left(5c_2^3-4c_2c_3\right)e_n^2d_n$$
$$+6\left(3c_2^2c_3-2c_2^4-c_2c_4\right)e_n^3d_n$$
$$+\left(144c_2^6-528c_2^4c_3+264c_2^3c_4+471c_2^2c_3^2\right.$$
$$-114c_2^2c_5-300c_2c_3c_4+42c_2c_6+33c_4^2$$
$$\left.\left.-12c_7-60c_3^3+60c_3c_5\right)e_n^6+O\left(e_n^7\right)\right].$$

Dividing equation (32) by equation (31), we get

$$\frac{\beta^2f(z_n)}{\left(\beta^2-\alpha\beta\right)f'(x_n)+\alpha\beta f'(y_n)} \quad (33)$$
$$=d_n\left[1+3c_2c_3e_n^3-c_2d_n+O\left(e_n^4\right)\right].$$

Since z_n is of order at least m, there exists a constant A such that

$$d_n=z_n-\xi=Ae_n^m+O\left(e_n^{m+1}\right). \quad (34)$$

Thus,

$$\begin{aligned} e_{n+1} &= z_n - \xi - \frac{\beta^2 f(z_n)}{(\beta^2 - \alpha\beta) f'(x_n) + \alpha\beta f'(y_n)} \\ &= d_n - d_n\left[1 + 3c_2c_3e_n^3 - c_2d_n + O\left(e_n^4\right)\right] \\ &= -3c_2c_3d_ne_n^3 + c_2d_n^2 + O\left(e_n^{n+4}\right). \end{aligned} \tag{35}$$

Hence, if $m < 3$ then using equations (34) and (35) reduce to

$$e_{n+1} = c_2A^2e_n^{2m} + O\left(e_n^{2m+1}\right), \tag{36}$$

which implies that the method defined by equation (17) is of order at least $2m$. Furthermore, if $m \geq 3$, then using equations (34) and (35), we obtain

$$e_{n+1} = -3c_2c_3Ae_n^{m+3} + O\left(e_n^{m+4}\right). \tag{37}$$

Therefore, the method defined by equation (17) is of order at least $m+3$.

This completes the proof.

3. The Concrete Iterative Methods

This section describes some interesting studies based on different form of $z_n = \phi(x_n; f(x_n), f'(x_n), f'(y_n))$. Throughout the rest of this article, y_n is defined by equation (21).

3.1. Some Fourth-Order Methods

Case 3.1. For the function ϕ defined by

$$\phi(x; f(x), f'(x), f'(y)) = x - \frac{f(x)}{f'(x)}, \tag{38}$$

then we obtain the two-step Newton's method

$$\begin{cases} z_n = x_n - \dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = z_n - \dfrac{f(z_n)}{f'(z_n)}. \end{cases} \tag{39}$$

It is easy to see that equation (39) is the well-known two-point fourth-order double-Newton method [12,13].

Case 3.2. If we take a second order method from [14]

$$z_n = x_n - \frac{f(x_n)}{f'(x_n) + \alpha' f(x_n)}, \tag{40}$$

where α' is a parameter, then we obtain the new four-order method

$$\begin{cases} z_n = x_n - \dfrac{f(x_n)}{f'(x_n) + \alpha' f(x_n)}, \\ x_{n+1} = z_n - \dfrac{\beta^2 f(z_n)}{(\beta^2 - \alpha\beta) f'(x_n) + \alpha\beta f'(y_n)}. \end{cases} \tag{41}$$

These two methods have order four and use four functional evaluations per step, and their efficiency index [15] is the same with the known methods of order two. However, numerical examples show that these modified methods may be efficient enough and have better performance as compared to the known methods of order two.

3.2. Some Sixth-Order Methods

Case 3.3. If we take a third-order variant of Newton's method appeared in [9]

$$z_n = x_n - \frac{2f(x_n)}{f'(y_n) + f'(x_n)}. \tag{42}$$

Then, we get the new sixth-order method

$$\begin{cases} z_n = x_n - \dfrac{2f(x_n)}{f'(y_n) + f'(x_n)}, \\ x_{n+1} = z_n - \dfrac{\beta^2 f(z_n)}{(\beta^2 - \alpha\beta) f'(x_n) + \alpha\beta f'(y_n)}. \end{cases} \tag{43}$$

Case 3.4. If we use the cubically convergent iterative scheme in [16]

$$x_{n+1} = x_n - \frac{f(x_n)(f'(x_n) + f'(y_n))}{2f'(x_n) f'(y_n)}. \tag{44}$$

Then, the following expressions can be resulted

$$\begin{cases} z_n = x_n - \dfrac{f(x_n)(f'(x_n) + f'(y_n))}{2f'(x_n) f'(y_n)}, \\ x_{n+1} = z_n - \dfrac{\beta^2 f(z_n)}{(\beta^2 - \alpha\beta) f'(x_n) + \alpha\beta f'(y_n)}. \end{cases} \tag{45}$$

The new methods (43) and (45) have the efficiency index equal to $6^{\frac{1}{4}} \approx 1.5651$, which is better than $\sqrt{2} \approx 1.4142$ of Newton's method and $3^{\frac{1}{3}} \approx 1.4422$ of the methods (42) and (44).

3.3. The Seventh-Order Method

Case 3.5. If we take the fourth-order Jarratt method [17] defined by

$$\begin{cases} y_n = x_n - \dfrac{2}{3}\dfrac{f(x_n)}{f'(x_n)}, \\ z_n = x_n - \dfrac{3f'(y_n) + f'(x_n)}{6f'(y_n) - 2f'(x_n)}\dfrac{f(x_n)}{f'(x_n)}. \end{cases} \tag{46}$$

Then, we obtain the new seventh-order method

$$\begin{cases} y_n = x_n - \dfrac{2}{3}\dfrac{f(x_n)}{f'(x_n)}, \\ z_n = x_n - \dfrac{3f'(y_n) + f'(x_n)}{6f'(y_n) - 2f'(x_n)}\dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = z_n - \dfrac{\beta^2 f(z_n)}{(\beta^2 - \alpha\beta) f'(x_n) + \alpha\beta f'(y_n)}. \end{cases} \tag{47}$$

Case 3.6. If we take the fourth-order method presented by Khattri and Abbasbandy in [18]

$$\begin{cases} y_n = x_n - \frac{2}{3}\frac{f(x_n)}{f'(x_n)}, \\ z_n = y_n - \left[1 + \left(\frac{21}{8}\right)\left(\frac{f'(y_n)}{f'(x_n)}\right) + \left(-\frac{9}{2}\right)\left(\frac{f'(y_n)}{f'(x_n)}\right)^2 \right. \\ \left. + \left(\frac{15}{8}\right)\left(\frac{f'(y_n)}{f'(x_n)}\right)^3\right]\frac{f(x_n)}{f'(x_n)}. \end{cases} \quad (48)$$

Then, we obtain the new seventh-order method

$$\begin{cases} y_n = x_n - \frac{2}{3}\frac{f(x_n)}{f'(x_n)}, \\ z_n = y_n - \left[1 + \left(\frac{21}{8}\right)\left(\frac{f'(y_n)}{f'(x_n)}\right) + \left(-\frac{9}{2}\right)\left(\frac{f'(y_n)}{f'(x_n)}\right)^2 \right. \\ \left. + \left(\frac{15}{8}\right)\left(\frac{f'(y_n)}{f'(x_n)}\right)^3\right]\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = z_n - \frac{\beta^2 f(z_n)}{\left(\beta^2 - \alpha\beta\right) f'(x_n) + \alpha\beta f'(y_n)}. \end{cases} \quad (49)$$

The methods defined by equations (47) and (49) require two function and two first derivative evaluations per iteration. Each method improves the order of convergence of the fourth-order method from four to seven with additional function evaluation at the point iterated by the fourth-order method. We have that the methods obtained by the formulae (47) and (49) have the efficiency index equal to $7^{\frac{1}{4}} \approx 1.6266$, which is better than the fourth-order methods $4^{\frac{1}{3}} \approx 1.5874$. It should be pointed out that many new higher-order methods can be obtained by considering the different choices of z_n in (17).

4. Numerical Examples

In this section, the results of some numerical tests are given to demonstrate the convergence efficiencies of various iterative schemes. We employ the present methods (39), (41) $(\alpha' = 1)$, (43), (45), (47) and (49) denoted by INW, IKT, IWM, IOM, IJA and IKA, respectively to solve some nonlinear equations and compare with Newton's method (NW), the method (KT) developed by Kanwar et al. [14], the method (WM) developed by Weerakoon et al. [9], the method (OM) by Özban [16], the Jarratt method (JA) and the method (KA) developed by Khattri et al. [18]. The equations $f(x) = 0$ was solved using the following test functions with corresponding starting values x_0:

$$f_1(x) = \sin^2 x - x^2 + 1, \xi_1^* = 1.4044916482153412260,$$

$$f_2(x) = x^2 - e^x - 3x + 2, \xi_2^* = 0.25753028543986076046,$$

$$f_3(x) = \cos x - xe^x + x^2, \xi_3^* = 0.63915409633200758106,$$

$$f_4(x) = \cos x - x, \xi_4^* = 0.73908513321516064166.$$

Numerical computations have been carried out using variable precision arithmetic, with 1000 digits, in Matlab 2013a. The stopping criterion is taken as $|x_{n+1} - x_n| + |f(x_{n+1})| < 10^{-100}$. In cases where the exact solution was not available, we used the approximation ξ^*, which was also calculated with 1000 digits. For simplicity, only 20 digits are displayed.

The computational order of convergence (COC) was given by (see [19])

$$COC = \frac{\ln\left(|x_{k+1} - x_k| / |x_k - x_{k-1}|\right)}{\ln\left(|x_k - x_{k-1}| / |x_{k-1} - x_{k-2}|\right)}. \quad (50)$$

Table 1 summarizes the results obtained by using the mentioned methods in order to estimate a root of nonlinear equations. For every function we specify the initial estimate x_0, the number of iterations N required to meet the stopping criteria, the value of $|f(x_{n+1})|$ in the last iteration and the value of COC.

We also compare our methods with the fourth-order King's method [20] (KM), the Ostrowski's method [21] (OST), the sixth order methods given by Chun and Ham [7] (CM1 and CM2), Parhi and Gupta [8] (PM), and the eighth order method given by Grau-Sánchez [22] (MG) (see Table 2).

Table 1. Numerical comparisons of the existing methods and the present combined iterative methods

	Method	N	$\lvert f(x_{n+1})\rvert$	CPU	**COC**
f_1, $x_0 = 1$					
	NW	9	3.4e-101	0.046	1.999851588919838
	INW	5	3.4e-101	0.031	3.991493528005224
	KT	11	3.7e-168	0.047	2.000000000152457
	IKT	6	1.6e-234	0.203	3.941604179745691
	WM	7	7.5e-266	0.088	3.045328591919475
	IWM	5	1.2e-566	0.111	6.262699932136188
	OM	5	1.1e-186	0.531	3.010588446816082

	Method	N	$\lvert f(x_{n+1}) \rvert$	CPU	COC
	IOM	3	6.5e-123	0.063	6.570253392709460
	JA	6	1.4e-334	0.152	4.248708984793318
	IJA	4	6.0e-426	0.060	6.883153998959128
	KA	8	3.0e-244	0.203	3.944947185376285
	IKA	6	9.0e-728	0.125	6.896899302326081
$f_1, x_0 = 2.3$					
	NW	9	1.7e-104	0.031	1.999762551535668
	INW	5	1.7e-104	0.031	3.992705876948916
	KT	10	6.9e-110	0.031	2.000000006728449
	IKT	6	3.8e-296	0.078	3.973931308562368
	WM	6	5.8e-106	0.031	2.977548164561699
	IWM	5	1.0e-520	0.046	5.428706327611860
	OM	6	1.3e-213	0.031	3.004705987760414
	IOM	4	1.7e-129	0.054	5.756054092345167
	JA	6	1.4e-311	0.076	3.991853110424756
	IJA	5	6.0e-426	0.078	6.910591531400821
	KA	6	1.0e-234	0.094	3.929232615276695
	IKA	5	1.9e-1000	0.124	6.991599899128005
$f_2, x_0 = 0$					
	NW	8	8.9e-201	0.031	2.001221658759151
	INW	5	1.9e-402	0.031	4.036830839017272
	KT	8	4.9e-124	0.046	2.000000001565404
	IKT	5	1.4e-289	0.047	4.072048404246373
	WM	5	7.8e-106	0.033	2.989730807986798
	IWM	4	5.4e-271	0.046	5.998509512335663
	OM	5	4.3e-112	0.172	2.999984519710846
	IOM	4	3.5e-276	0.125	5.992519937559335
	JA	5	1.0e-286	0.057	3.987739193299696
	IJA	4	2.8e-827	0.061	6.997013442884977
	KA	5	1.6e-292	0.109	4.220027174017621
	IKA	4	5.3e-833	0.094	7.095300609185996
$f_2, x_0 = 1$					
	NW	8	1.7e-189	0.032	1.997942151132725
	INW	5	7.1e-380	0.032	3.725609846208910
	KT	9	4.1e-138	0.031	2.000000000185425
	IKT	5	2.7e-207	0.063	3.878358818775769
	WM	6	1.4e-201	0.062	3.038062042190221
	IWM	4	1.2e-201	0.031	5.825290885992024
	OM	6	3.4e-206	0.047	3.001050663973445
	IOM	4	3.1e-202	0.047	5.843786348256205
	JA	5	2.4e-258	0.062	3.722160566330791
	IJA	4	3.2e-633	0.062	6.833672609585321
	KA	5	5.6e-264	0.110	3.981598688393819
	IKA	4	5.5e-643	0.110	7.023459826814717
$f_3, x_0 = 1$					
	NW	9	1.3e-151	0.047	2.001106997169595
	INW	5	1.3e-151	0.031	3.999685578348938
	KT	9	1.3e-108	0.062	2.000000012840528

	Method	N	$\left\|f\left(x_{n+1}\right)\right\|$	CPU	COC
	IKT	5	7.2e-122	0.093	3.998627364224366
	WM	6	2.9e-131	0.063	3.000546669049605
	IWM	4	3.9e-133	0.062	5.866966779516045
	OM	6	3.4e-186	0.234	3.000513674703602
	IOM	4	2.8e-129	0.062	5.983351099780604
	JA	6	3.4e-425	0.109	3.990537088703305
	IJA	4	9.3e-444	0.106	6.985005730502670
	KA	6	3.7e-302	0.172	3.991952875605968
	IKA	5	0	0.125	6.825746339976237
f_3, $x_0 = 0.5$					
	NW	8	8.9e-122	0.046	2.003980933345205
	INW	5	2.6e-243	0.047	4.016633132132276
	KT	9	8e-184	0.047	1.999999964933080
	IKT	5	7.0e-205	0.078	4.031322751779069
	WM	6	8.7e-214	0.054	3.044070100120532
	IWM	4	5.2e-203	0.078	6.026855559538948
	OM	6	2.7e-292	0.062	2.999983735018816
	IOM	4	1.6e-217	0.062	5.996940163017289
	JA	5	5.6e-257	0.085	3.995556784204996
	IJA	4	4.7e-723	0.101	6.971215155322127
	KA	6	3.2e-400	0.141	4.116625589705837
	IKA	4	4.8e-552	0.140	6.989197183612720
f_4, $x_0 = 0$					
	NW	9	1.2e-166	0.016	1.998849110827749
	INW	5	1.2e-166	0.031	3.998682264532630
	KT	11	3.0e-142	0.047	2.000000000043006
	IKT	6	3.1e-220	0.046	3.848261256965063
	WM	6	4.6e-189	0.031	3.579101980170624
	IWM	4	4.5e-124	0.031	6.409937874573192
	OM	6	3.1e-180	0.031	2.998710147557952
	IOM	4	4.0e-183	0.047	6.182149360844007
	JA	6	1.2e-388	0.047	3.967349085468574
	IJA	4	3.7e-477	0.046	6.909444215467527
	KA	7	4.9e-435	0.094	3.609092926736953
	IKA	5	1.0e-851	0.109	7.092117798320114
f_4, $x_0 = 1.7$					
	NW	8	4.0e-130	0.031	1.994301682099229
	INW	5	2.0e-260	0.047	3.594716721885800
	KT	10	5.7e-177	0.031	2.000000008480267
	IKT	5	6.9e-190	0.046	3.999745340001081
	WM	6	1.2e-196	0.047	3.150905120339132
	IWM	4	7.4e-148	0.032	5.930006529346190
	OM	6	1.5e-177	0.031	3.001592484975891
	IOM	4	6.2e-143	0.031	5.886505865949367
	JA	6	1.4e-443	0.062	3.665874280525460
	IJA	4	1.7e-484	0.054	6.799062297633464
	KA	6	7.0e-428	0.078	3.868210819761995
	IKA	4	1.5e-466	0.093	6.808033648694732

Table 2. Numerical comparisons of the methods of KM, OST, PM, CM1, CM2, MG and the present methods

	Method	N	$\lvert f(x_{n+1})\rvert$	CPU
$f_1,\ x_0=1$				
	INW	5	3.4e-101	0.031
	IKT	6	1.6e-234	0.203
	KM	10	6.3e-301	0.251
	OST	5	1.0e-109	0.062
	IWM	5	1.2e-566	0.111
	IOM	3	6.5e-123	0.063
	CM1	5	1.4e-226	0.098
	CM2	6	3.6e-245	0.090
	PM	5	1.2e-566	0.068
	IJA	4	6.0e-426	0.060
	IKA	6	9.0e-728	0.125
	MG	5	7.6e-345	0.121
$f_1, x_0=2.3$				
	INW	5	1.7e-104	0.031
	IKT	6	3.8e-296	0.078
	KM	6	1.0e-259	0.078
	OST	6	8.3e-389	0.063
	IWM	5	1.0e-520	0.046
	IOM	4	1.7e-129	0.054
	CM1	5	1.6e-391	0.115
	CM2	5	1.4e-372	0.085
	PM	5	1.0e-520	0.075
	IJA	5	6.0e-426	0.078
	IKA	5	1.9e-1000	0.124
	MG	4	1.6e-130	0.320
$f_2,\ x_0=0$				
	INW	5	1.9e-402	0.031
	IKT	5	1.4e-289	0.047
	KM	5	1.9e-376	0.059
	OST	5	1.1e-352	0.063
	IWM	4	5.4e-271	0.046
	IOM	4	3.5e-276	0.125
	CM1	4	8.2e-313	0.081
	CM2	4	4.2e-291	0.067
	PM	4	5.4e-271	0.052
	IJA	4	2.8e-827	0.061
	IKA	4	5.3e-833	0.094
	MG	4	1.5e-387	0.133
$f_2,\ x_0=1$				
	INW	5	7.1e-380	0.032
	IKT	5	2.7e-207	0.063
	KM	5	3.8e-281	0.080
	OST	5	6.6e-258	0.047
	IWM	4	1.2e-201	0.031
	IOM	4	3.1e-202	0.047
	CM1	4	3.4e-194	0.074
	CM2	4	1.4e-194	0.062
	PM	4	1.2e-201	0.059
	IJA	4	3.2e-633	0.062
	IKA	4	5.5e-643	0.110
	MG	4	3.5e-326	0.129

	Method	N	$\left\| f\left(x_{n+1}\right)\right\|$	CPU
f_3, $x_0 = 1$				
	INW	5	1.3e-151	0.031
	IKT	5	7.2e-122	0.093
	KM	5	1.5e-103	0.196
	OST	5	9.5e-187	0.094
	IWM	4	3.9e-133	0.062
	IOM	4	2.8e-129	0.062
	CM1	4	9.0e-124	0.102
	CM2	4	4.3e-118	0.098
	PM	4	4.0e-133	0.131
	IJA	4	9.3e-444	0.106
	IKA	5	0	0.125
	MG	4	1.1e-205	0.190
f_3, $x_0 = 0.5$				
	INW	5	2.6e-243	0.047
	IKT	5	7.0e-205	0.078
	KM	5	1.1e-163	0.124
	OST	5	1.0e-292	0.062
	IWM	4	5.2e-203	0.078
	IOM	4	1.6e-217	0.062
	CM1	4	1.2e-200	0.120
	CM2	4	7.4e-188	0.096
	PM	4	5.3e-203	0.093
	IJA	4	4.7e-723	0.101
	IKA	4	4.8e-552	0.140
	MG	4	2.0e-316	0.203
f_4, $x_0 = 0$				
	INW	5	1.2e-166	0.031
	IKT	6	3.1e-220	0.046
	KM	6	1.8e-197	0.057
	OST	5	5.5e-141	0.031
	IWM	4	4.5e-124	0.031
	IOM	4	4.0e-183	0.047
	CM1	5	3.1e-489	0.073
	CM2	5	1.6e-373	0.055
	PM	4	4.5e-124	0.038
	IJA	4	3.7e-477	0.046
	IKA	5	1.0e-851	0.109
	KLW	4		
	MG	4	8.8e-127	0.086
f_4, $x_0 = 1.7$				
	INW	5	2.0e-260	0.047
	IKT	5	6.9e-190	0.046
	KM	5	6.8e-178	0.045
	OST	5	4.4e-192	0.031
	IWM	4	7.4e-148	0.032
	IOM	4	6.2e-143	0.031
	CM1	4	1.5e-141	0.067
	CM2	4	1.0e-139	0.051
	PM	4	7.3e-148	0.044
	IJA	4	1.7e-484	0.054
	IKA	4	1.5e-466	0.093
	MG	4	7.2e-244	0.089

In Table 1 it is seen that our combined iterative methods generally arrive at the iterated solution with less number of iterations than the corresponding second, cubic and the fourth methods, so that the proposed methods improve the computational efficiency of the existing iterative methods. Our examples show that the combined iterative methods sometimes require more CPU time per iteration, compared to the existing methods. Although our methods require more time per iteration, they yield better numerical results. The numerical results in Table 1 show that for almost all of the test functions, our methods are well in accordance with the theory developed in section 2.

The test results in Table 2 show that for most of the functions we tested, the methods introduced in the present presentation for numerical tests have equal or better performance compared to the other methods of the same order. In each of these 8 test cases, the INW method outperformed the KM method in every case and it outperformed the OST method in 5 out of the 8 cases. Our IKT method outperformed KM method in 6 out of the 8 cases. We also implemented the three 6th order schemes of [7,8] using these 8 cases, and found that the IWM and IOM methods outperformed the PM, CM1 and CM2 methods in 7 out of the 8 cases.

In [22] a eighth-order method, denoted with MG was considered. The IJA method outperformed the MG method in 7 out of the 8 cases, and the IKA method outperformed the MG method in 4 out of the 8 cases. Besides, we can see that the local convergence property of the new methods depending on the structure of the tested functions and the choice of initial approximations.

5. Conclusion

The new modified Newton's methods presented in this paper offer an increase rate of convergence over the existing methods. Unlike other higher-order methods, the distinct feature of such methods is only to add the evaluation of the function at another point, while their order of convergence can be improved effectively. Our new combined iterative methods are relatively simple and robust, more high-order convergence methods can be constructed by using the family of methods (17).

Computational results for test functions, presented in Table 1 and Table 2, show that our methods are efficient and show at least equal or better performance as compared with other higher order (4th order, 6th order and 8th order) schemes or Newton's method itself. Our methods show similar good performance for other functions.

Acknowledgements

The authors gratefully acknowledge the support of the National Natural Science Foundations of China (NNSFC) through grants No.11302184, 11202189, the Scientific Research Fund of Zhejiang Provincial Education Department through grant No. Y201121157 and the Scientific Research Foundation of Xiamen University of Technology through grant No. YKJ13007R.

References

[1] Zhou, X.J., "A class of Newton's methods with third-order convergence," *Applied Mathematics Letters*, 20 (9). 1026-1030. Sep.2007.

[2] Cordero, A., Jordán, C. and Torregrosa, J.R.,"One-point Newton-type iterative methods: A unified point of view," *Journal of Computational and Applied Mathematics*, 275. 366-374. Feb. 2015.

[3] Neta, B. and Scott, M., "On a family of Halley-like methods to find simple roots of nonlinear equations," *Applied Mathematics and Computation*, 219 (15). 7940-7944. Mar. 2013.

[4] Soleymani, F., Khattri, S.K. and Vanani, S.K., "Two new classes of optimal Jarratt-type fourth-order methods," *Applied Mathematics Letters*, 25 (5). 847-853. May. 2012.

[5] Wang, X. and Liu, L.P., "Two new families of sixth-order methods for solving non-linear equations," *Applied Mathematics and Computation*, 213 (1). 73-78. Jul. 2009.

[6] Kou, J.S., Li, Y.T. and Wang, X.H., "Some modifications of Newton's method with fifth-order convergence," *Journal of Computational and Applied Mathematics*, 209 (2). 146-152. Dec. 2007.

[7] Chun, C. and Ham, Y., "Some sixth-order variants of Ostrowski root-finding methods, "*Applied Mathematics and Computation*, 193 (2). 389-394. Nov. 2007.

[8] Parhi, S.K. and Gupta, D.K., "A sixth order method for nonlinear equations," *Applied Mathematics and Computation*, 203 (1). 50-55. Sep. 2008.

[9] Weerakoon, S. and Fernando, T.G.I., "A variant of Newton's method with accelerated third-order convergence," *Applied Mathematics Letters*, 13 (8). 87-93. Nov. 2000.

[10] Petkovic, S.M., Neta B., Petkovic L.D. and Džunic, J., "Multiple methods for solving nonlinear equations: A survey," *Applied Mathematics and Computation*, 226. 635-660. Jan. 2014.

[11] Steffensen, I.F., "Remarks on iteration," *Scandinavian Actuarial Journal*, 1993(1). 64-72. Dec. 2011.

[12] Khattri, S.K. and Argyros, I.K., "How to develop fourth and seventh order iterative methods," *Novi Sad Journal of Mathematics*, 40 (2). 61-67. 2010.

[13] Traub, J.F., *Iterative methods for the solution of equations*, Chelsea Publishing Company, New York, 1982.

[14] Kanwar, V. and Tomar, S.K., "Modified families of Newton, Halley and Chebyshev methods," *Applied Mathematics and Computation*, 192 (1). 20-26. Sep. 2007.

[15] Gautschi, W., *Numerical Analysis: An introduction*, Birkhäuser, 1997.

[16] Özban, A. Y., "Some new variants of Newton's method," *Applied Mathematics Letters*, 17 (6). 677-682. Jun. 2004.

[17] Jarratt, P., "Some efficient fourth order multipoint methods for solving equations," *BIT Numerical Mathematics*, 9(2). 119-124. Jun. 1969.

[18] Khattri, S.K. and Abbasbandy, S., "Optimal fourth order family of iterative methods," *Matematicki Vesnik*, 63 (1). 67-72. Jan. 2011.

[19] Cordero, A. and Torregrosa, J.R., "Variants of Newton's method using fifth-order quadrature formulas," *Applied Mathematics and Computation*, 190 (1). 686-698. Jul. 2007.

[20] King, R.F, "A family of fourth-order methods for nonlinear equations," *SIAM Journal on Numerical Analysis*, 10 (5). 876-879. Oct. 1973.

[21] Ostrowski, A.M., *Solution of equations in Euclidean and Banach spaces*, Academic, New York, 1960.

[22] Grau-Sánchez, M., "Improving order and efficiency: Composition with a modified Newton's method," *Journal of Computational and Applied Mathematics*, 231 (2). 592-597. Sep. 2009.