NEW TWELFTH ORDER ALGORITHMS FOR SOLVING NONLINEAR EQUATIONS BY USING VARIATIONAL ITERATION TECHNIQUE

MUHAMMAD NAWAZ[1], AMIR NASEEM[2], WAQAS NAZEER[3]

Abstract. In this paper, we proposed three new algorithms for solving non-linear equations by using variational iteration technique. We discuss the convergence criteria of our newly developed algorithms. To demonstrate the efficiency and performance of these methods, several numerical examples are given which show that our generated methods are best as compared to Newton's method, Halley's method, Househölder's method and other well known iterative methods. The variational iteration technique can be used to suggest a wide class of new iterative methods for solving a system of non-linear equations.

Key words : Non-linear equations, Newton's method, Halley's method, Househölder's method.
AMS SUBJECT : 26A51, 26A33, 33E12.

1. Introduction

Most of the problems in mathematics, physics and engineering sciences is related to non-linear equation of the form

$$f(x) = 0.$$

The solution of such type of non-linear equations cannot be find directly except in special cases. Therefore we have to adopt iterative methods for

---

[1]Department of Mathematics, University of Lahore, Pakpattan Campus, Pakpattan Pakistan. Email: mathvision204@gmail.com
[2]Department of Mathematics, University of Management and Technology, Lahore 54000, Pakistan. Email: amir14514573@yahoo.com
[3]Division of Science and Technology, University of Education, Lahore Pakistan. Email: nazeer.waqas@ue.edu.pk.

solving such type of equations. In iterative methods, we start with an initial guess x_0 which is improved step by step by means of iterations.In recent years, a large number of iterative methods have been developed using different techniques such as decomposition method, Taylor's series, perturbation method, quadrature formulas and variational iteration technique [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18] and the references therein. The most famous method for solving non-linear equations is Newton's method which can be written as:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}. \tag{1}$$

This is so-called the Newton's method [12].

To improve its convergence, various modified methods has been developed in literature. Some of them are given in [3, 8, 9, 10, 11, 19, 20], and the references therein.

In this paper, we proposed three new algorithms by applying variational iteration technique by considering two auxiliary functions $\phi(x)$ and $\psi(x)$. The first one, $\phi(x)$ acts as predictor function having convergence order q where $q \geq 1$ and the second function $\psi(x)$ acts as corrector function having convergence order r with $r \geq 1$. The predictor function helps to obtain iterative methods of convergence order $q + r$. Using variational iteration technique, we develop new iterative methods with higher order of convergence. The variational iteration technique was introduced by Inokuti et al [13]. Using this technique, Noor and Shah [17] derived some iterative methods for solving the non-linear equations.The purpose of this technique was to solve a variety of diverse problems [14, 15, 16] .

Now we apply the described technique to obtain higher-order iteraive methods. New methods are very fast using less number of iterations to reach the required solution, free from 3rd and higher derivatives with twelfth order of convergence which raises the efficiency index of these methods. We also discuss the convergence criteria of these new iterative methods. Several examples are given to show the performance of our proposed methods as compare to the other similar existing methods.

2. Construction of Iterative Methods Using Variational Technique

We use the variational iteration technique to derive some new iterative methods. These are multi-step methods consisting of predictor and corrector steps. The convergence of our methods is better than the one-step methods. The variational iteration technique is used to obtain some new iterative methods of order $q + r$ where $q, r \geq 1$ is the order of convergence of the auxiliary iteration functions $\phi(x)$ and $\psi(x)$. Now consider the non-linear equation of

the form

$$f(x) = 0 \tag{2}$$

Suppose that α is the simple root and γ is the initial guess sufficiently close to α. For the sake of convenience and to give the basic idea, we consider the approximate solution x_n of (2) such that

$$f(x_n) \neq 0.$$

We consider $\phi(x)$ and $\psi(x)$ as two iteration functions of order q and r respectively. Then

$$x_{n+1} = \phi(x_n) + \mu[f(\psi(x_n)g(\psi(x_n)]^t, \tag{3}$$

where $t = \frac{q}{r}$ is a recurrence relation which generates iterative methods of order $q+r$ and $g(x)$ is any arbitrary function which later on is converted to $g(\psi(x_n)$ and μ is a parameter which is usually called the Lagrange's multiplier and can be identified by the optimality condition. Using the optimality criteria from (3), we have

$$\mu = -\frac{\phi'(x_n)[f(\psi(x_n))g(\psi(x_n))]^{1-t}}{t\psi'(x_n)[f'(\psi'(x_n))g(\psi(x_n)) + f(\psi(x_n))g'(\psi'(x_n))]}. \tag{4}$$

From (3) and (4), we get

$$x_{n+1} = \phi(x_n) - \frac{\phi'(x_n)}{t\psi'(x_n)} \frac{f(\psi(x_n))g(\psi(x_n))}{[f'(\psi(x_n))g(\psi(x_n)) + f(\psi(x_n))g'(\psi(x_n))]}. \tag{5}$$

Now we are going to apply eq(5) for constructing a general iterative scheme for iterative methods. For this, suppose that

$$\psi(x_n) = y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}, \tag{6}$$

which is well known Househölder's method with cubic convergence. With the help of (5) and (6), we can write

$$x_{n+1} = \phi(x_n) - \frac{\phi'(x_n)f(y_n)g(y_n)}{ty'_n[f'(y_n)g(y_n) + f(y_n)g'(y_n)]}. \tag{7}$$

Let

$$\phi(x_n) = z_n = y_n - \frac{f(y_n)}{f'(y_n)} - \frac{f^2(y_n)f''(y_n)}{2f'^3(y_n)}, \tag{8}$$

where $y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}$

Which is two step Househölder's method having convergence of order nine. Differentiate eq.(8) w.r.t " x " we have

$$\phi'(x_n) = z'_n = \frac{f^2(y_n)[3f''^2(y_n) - f'(y_n)f'''(y_n)]}{2f'^4(y_n)}y'_n, \tag{9}$$

and from Taylor's series for $f''(z_n)$, we can write

$$f''(z_n) = f''(y_n) + (y_n - z_n)f'''(y_n) + \dots \tag{10}$$

Ignoring square and higher powers of $(y_n - z_n)$, we get

$$f''(z_n) \approx f''(y_n) + (y_n - z_n)f'''(y_n). \tag{11}$$

With the help of (9) and (11), we get

$$\phi'(x_n) = \frac{f(y_n)[2f'^4(y_n)f''(z_n) - 2f'^4(y_n)f''(y_n) + 6f(y_n)f'^2(y_n)f''^2(y_n) + 3f''^3(y_n)f^2(y_n)]}{2f'^4(y_n)[2f'^2(y_n) + f(y_n)f''(y_n)]}y_n'. \tag{12}$$

Using (11) in (7), we obtain

$$x_{n+1} = z_n - \frac{f^2(y_n)g(y_n)[2f'^4(y_n)f''(z_n) - 2f'^4(y_n)f''(y_n) + 6f(y_n)f'^2(y_n)f''^2(y_n) + 3f''^3(y_n)f^2(y_n)]}{2tf'^4(y_n)[2f'^2(y_n) + f(y_n)f''(y_n)][f'(y_n)g(y_n) + f(y_n)g'(y_n)]}. \tag{13}$$

Here $t = \frac{9}{3} = 3$, which is according to the above described technique. Then eq.(13) becomes:

$$x_{n+1} = z_n - \frac{f^2(y_n)g(y_n)[2f'^4(y_n)f''(z_n) - 2f'^4(y_n)f''(y_n) + 6f(y_n)f'^2(y_n)f''^2(y_n) + 3f''^3(y_n)f^2(y_n)]}{6f'^4(y_n)[2f'^2(y_n) + f(y_n)f''(y_n)][f'(y_n)g(y_n) + f(y_n)g'(y_n)]}. \tag{14}$$

Relation(14) is the main and general iterative scheme, which we use to deduce iterative methods for solving non-linear equations by considering some special cases of the auxiliary functions g.

2.1. **Case 1.** Let $g(x_n) = \exp(-\beta x_n)$, then $g'(x_n) = -\beta g(x_n)$. Using these values in (14), we obtain the following algorithm.

Algorithm 1. *For a given x_0, compute the approximate solution x_{n+1} by the following iterative schemes:*

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}, n = 0, 1, 2, \dots,$$

$$z_n = y_n - \frac{f(y_n)}{f'(y_n)} - \frac{f^2(y_n)f''(y_n)}{2f'^3(y_n)}$$

$$x_{n+1} = z_n - \frac{f^2(y_n)[2f'^4(y_n)f''(z_n) - 2f'^4(y_n)f''(y_n) + 6f(y_n)f'^2(y_n)f''^2(y_n) + 3f''^3(y_n)f^2(y_n)]}{6f'^4(y_n)[2f'^2(y_n) + f(y_n)f''(y_n)][f'(y_n) - \beta f(y_n)]}.$$

2.2. **Case 2.** Let $g(x_n) = \exp(-\beta f(x_n))$, then $g'(x_n) = -\beta f'(x_n)g(x_n)$. Using these values in (14), we obtain the following algorithm.

Algorithm 2. *For a given x_0, compute the approximate solution x_{n+1} by the following iterative schemes:*

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}, n = 0, 1, 2, \dots,$$

$$z_n = y_n - \frac{f(y_n)}{f'(y_n)} - \frac{f^2(y_n)f''(y_n)}{2f'^3(y_n)}$$

$$x_{n+1} = z_n - \frac{f^2(y_n)[2f'^4(y_n)f''(z_n) - 2f'^4(y_n)f''(y_n) + 6f(y_n)f'^2(y_n)f''^2(y_n) + 3f''^3(y_n)f^2(y_n)]}{6f'^4(y_n)[2f'^2(y_n) + f(y_n)f''(y_n)][f'(y_n) - \beta f(y_n)f'(y_n)]}.$$

2.3. **Case 3.** Let $g(x_n) = \exp(-\frac{\beta}{f'(x_n)})$, then $g'(x_n) = \beta\frac{f''(x_n)}{f'^2(x_n)}g(x_n)$. Using these values in (14), we obtain the following algorithm.

Algorithm 3. *For a given x_0, compute the approximate solution x_{n+1} by the following iterative schemes:*

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}, n = 0, 1, 2, ...,$$

$$z_n = y_n - \frac{f(y_n)}{f'(y_n)} - \frac{f^2(y_n)f''(y_n)}{2f'^3(y_n)}$$

$$x_{n+1} = z_n - \frac{f^2(y_n)f'^2(y_n)[2f'^4(y_n)f''(z_n) - 2f'^4(y_n)f''(y_n) + 6f(y_n)f'^2(y_n)f''^2(y_n) + 3f''^3(y_n)f^2(y_n)]}{6f'^4(y_n)[2f'^2(y_n) + f(y_n)f''(y_n)][f'^3(y_n) + \beta f(y_n)f''(y_n)]}.$$

To obtain best results in all above algorithms, always choose that values of β which makes the denominator non-zero and largest in magnitude.

3. Convergence Analysis

In this section, we discuss the convergence order of the main and general iteration scheme (14).

Theorem 1. *Suppose that α is a root of the equation $f(x) = 0$. If $f(x)$ is sufficiently smooth in the neighborhood of α, then the convergence order of the main and general iteration scheme, described in relation (14) is atleast twelve.*

Proof. To analysis the convergence of the main and general iteration scheme, described in relation (14), suppose that α is a root of the equation $f(x) = 0$ and e_n be the error at nth iteration, then $e_n = x_n - \alpha$ and by using Taylor series expansion, we have

$$f(x) = f'(\alpha)e_n + \frac{1}{2!}f''(\alpha)e_n^2 + \frac{1}{3!}f'''(\alpha)e_n^3 + \frac{1}{4!}f^{(iv)}(\alpha)e_n^4 + \frac{1}{5!}f^{(v)}(\alpha)e_n^5$$
$$+ \frac{1}{6!}f^{(vi)}(\alpha)e_n^6 + ... + O(e_n^{13}).$$

$$f(x) = f'(\alpha)[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + c_5e_n^5 + c_6e_n^6 + ... + O(e_n^{13})]. \quad (15)$$

$$f'(x_n) = f'(\alpha)[1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + 5c_5e_n^4 + 6c_6e_n^5 + 7c_7e_n^6$$
$$+ ... + O(e_n^{13})]. \quad (16)$$

$$f''(x_n) = f'(\alpha)[2c_2 + 6c_3e_n + 12c_4e_n^2 + 20c_5e_n^3 + 30c_6e_n^4 + 42c_7e_n^5 + 56c_8e_n^6$$
$$+ ... + O(e_n^{13})]. \quad (17)$$

Where
$$c_n = \frac{1}{n!}\frac{f^{(n)}(\alpha)}{f'(\alpha)}.$$

With the help of 3.1, 3.2 and 3.3, we get

$$
\begin{aligned}
y_n &= \alpha + (-c_3 + 2c_2^2)e_n^3 + (12c_2c_3 - 9c_2^3 - 3c_4)e_n^4 + (15c_3^2 - 63c_3c_2^2 \\
&\quad + 30c_2^4 + 24c_2c_4 - 6c_5)e_n^5 + (-10c_6 + 40c_2c_5 + 55c_4c_3 - 112c_4c_2^2 \\
&\quad - 136c_2c_3^2 + 251c_3c_2^3 - 88c_2^5)e_n^6 + ... + O(e_n^{13}).dfq \qquad (18)
\end{aligned}
$$

$$
\begin{aligned}
f(y_n) &= f'(\alpha)[-c_3 + 2c_2^2)e_n^3 + (12c_2c_3 - 9c_2^3 - 3c_4)e_n^4 + (15c_3^2 \\
&\quad - 63c_3c_2^2 + 30c_2^4 + 24c_2c_4 - 6c_5)e_n^5 + (-10c_6 + 40c_2c_5 + 55c_4c_3 \\
&\quad - 112c_4c_2^2 - 135c_2c_3^2 + 247c_3c_2^3 - 84c_2^5)e_n^6 + ... + O(e_n^{13})]. \qquad (19)
\end{aligned}
$$

$$
\begin{aligned}
f'(y_n) &= f'(\alpha)[1 + (-2c_2c_3 + 4c_2^3)e_n^3 + (24c_3c_2^2 \\
&\quad - 18c_2^4 - 6c_2c_4)e_n^4 + (30c_2c_3^2 - 126c_3c_2^3 + 60c_2^5 + 48c_4c_2^2 - 12c_2c_5)e_n^5 \\
&\quad + (-20c_2c_6 + 80c_2^2c_5 + 110c_2c_4c_3 - 224c_4c_2^3 - 284c_2^2c_3^2 + 514c_3c_2^4 \\
&\quad - 176c_2^6 + 3c_3^3)e_n^6 + ... + O(e_n^{13})]. \qquad (20)
\end{aligned}
$$

$$
\begin{aligned}
f''(y_n) &= f'(\alpha)[2c_2 + (-6c_3^2 + 12c_3c_2^2)e_n^3 + (72c_2c_3^2 - 54c_3c_2^3 - 18c_4c_3)e_n^4 \\
&\quad + (90c_3^3 - 378c_2^2c_3^2 + 180c_3c_2^4 + 144c_2c_4c_3 - 36c_3c_5)e_n^5 + (-60c_3c_6 \\
&\quad + 240c_3c_2c_5 + 342c_4c_3^2 - 720c_3c_4c_2^2 - 816c_2c_3^3 + 1506c_3^2c_2^3 - 528c_3c_2^5 \\
&\quad + 48c_4c_2^4)e_n^6 + ... + O(e_n^{13})]. \qquad (21)
\end{aligned}
$$

$$
\begin{aligned}
g(y_n) &= g(\alpha) + g'(\alpha)(-c_3 + 2c_2^2)e_n^3 + g'(\alpha)(12c_2c_3 - 9c_2^3 - 3c_4)e_n^4 \\
&\quad + g'(\alpha)(15c_3^2 - 63c_3c_2^2 + 30c_2^4 + 24c_2c_4 - 6c_5)e_n^5 \\
&\quad + [g'(\alpha)(-10c_6 + 40c_2c_5 + 55c_4c_3 - 112c_4c_2^2 - 136c_2c_3^2 \\
&\quad + 251c_3c_2^3 - 88c_2^5) + 2g''(\alpha)(-\frac{c_3}{2} + c_2^2)^2]e_n^6 + ... + O(e_n^{13}). \qquad (22)
\end{aligned}
$$

$$
\begin{aligned}
g'(y_n) &= g'(\alpha) + g''(\alpha)(-c_3 + 2c_2^2)e_n^3 + g''(\alpha)(12c_2c_3 - 9c_2^3 - 3c_4)e_n^4 \\
&\quad + g''(\alpha)(15c_3^2 - 63c_3c_2^2 + 30c_2^4 + 24c_2c_4 - 6c_5)e_n^5 + [g''(\alpha)(-10c_6 \\
&\quad + 40c_2c_5 + 55c_4c_3 - 112c_4c_2^2 - 136c_2c_3^2 \\
&\quad + 251c_3c_2^3 - 88c_2^5) + 2g''(\alpha)(-\frac{c_3}{2} + c_2^2)^2]e_n^6 + ... + O(e_n^{13}). \qquad (23)
\end{aligned}
$$

With the help of $(3.4 - 3.7)$, we have

$$
z_n = \alpha + (-8c_2^2c_3^3 + 24c_3^2c_2^4 - 32c_3c_2^6 + c_3^4 + 16c_2^8)e_n^9 + ... + O(e_n^{13}). \qquad (24)
$$

$$
\begin{aligned}
f(z_n) &= f'(\alpha)[(-8c_2^2c_3^3 + 24c_3^2c_2^4 - 32c_3c_2^6 + c_3^4 + 16c_2^8)e_n^9 + ... \\
&\quad + O(e_n^{13})]. \qquad (25)
\end{aligned}
$$

$$f'(z_n) = f'(\alpha)[1 + (-16c_3^3c_2^3 + 48c_3^2c_2^5 - 64c_3c_2^7 + 2c_2c_3^4 + 32c_2^9)e_n^9$$
$$+ ... + O(e_n^{13})]. \tag{26}$$

$$f''(z_n) = f'(\alpha)[2c_2 + (-48c_2^2c_3^4 + 144c_3^3c_2^4 - 192c_3^2c_2^6 + 6c_3^5 + 96c_3c2^8)e_n^9$$
$$+ ... + O(e_n^{13})]. \tag{27}$$

Using equations $(3.1 - 3.13)$ in general iteration scheme(14), we get the same result as given below

$$x_{n+1} = \alpha + \frac{80(-\frac{c_3}{2} + c_2^2)^4[(c_2^3 - c_2c_3 - \frac{c_4}{5})g(\alpha) - \frac{2}{5}g'(\alpha)(-\frac{1}{2}c_3 + c_2^2)]}{g(\alpha)}e_n^{12} + O(e_n^{13}).$$

Which implies that

$$e_{n+1} = \frac{80(-\frac{c_3}{2} + c_2^2)^4[(c_2^3 - c_2c_3 - \frac{c_4}{5})g(\alpha) - \frac{2}{5}g'(\alpha)(-\frac{1}{2}c_3 + c_2^2)]}{g(\alpha)}e_n^{12} + O(e_n^{13}),$$

Which implies that the main and general iteration scheme(14) is of twelfth order of convergence and all algorithms deduced from it have also twelve order of convergence.

3.1. alternative. To analysis the convergence of the main and general iteration scheme, described in relation (14), suppose that α is a root of the equation $f(x) = 0$ and e_n be the error at nth iteration, then $e_n = x_n - \alpha$ and by using Taylor series expansion, we have

$$f(x) = f'(\alpha)e_n + \frac{1}{2!}f''(\alpha)e_n^2 + \frac{1}{3!}f'''(\alpha)e_n^3 + \frac{1}{4!}f^{(iv)}(\alpha)e_n^4 + ... + O(e_n^{13})$$

$$f(x) = f'(\alpha)[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + ... + O(e_n^{13})] \tag{28}$$

$$f'(x_n) = f'(\alpha)[1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + .. + O(e_n^{13})] \tag{29}$$

$$f''(x_n) = f'(\alpha)[2c_2 + 6c_3e_n + 12c_4e_n^2 + 20c_5e_n^3 + ... + O(e_n^{13})] \tag{30}$$

Where

$$c_n = \frac{1}{n!}\frac{f^{(n)}(\alpha)}{f'(\alpha)}$$

With the help of 3.12, 3.13 and 3.14, we get

$$y_n = \alpha + (-c_3 + 2c_2^2)e_n^3 + (12c_2c_3 - 9c_2^3 - 3c_4)e_n^4 + ... + O(e_n^{13}) \tag{31}$$

$$f(y_n) = f'(\alpha)[(y_n - \alpha) + c_2(y_n - \alpha)^2 + c_3(y_n - \alpha)^3 + c_4(y_n - \alpha)^4$$
$$+ O(e_n^{13})] \tag{32}$$

$$f'(y_n) = f'(\alpha)[1 + 2c_2(y_n - \alpha) + 3c_3(y_n - \alpha)^2 + 4c_4(y_n - \alpha)^3$$
$$+ 5c_5(y_n - \alpha)^4 + O(e_n^{13})] \tag{33}$$

$$
\begin{aligned}
f''(y_n) &= f'(\alpha)[2c_2 + 6c_3(y_n - \alpha) + 12c_4(y_n - \alpha)^2 + 20c_5(y_n - \alpha)^3 \\
&+ 30c_6(y_n - \alpha)^4 + O(e_n^{13})]
\end{aligned}
\tag{34}
$$

$$
\begin{aligned}
g(y_n) &= g(\alpha) + (y_n - \alpha)g'(\alpha) + \frac{(y_n - \alpha)^2}{2!}g''(\alpha) + \frac{(y_n - \alpha)^3}{3!}g'''(\alpha) \\
&+ \frac{(y_n - \alpha)^4}{4!}g^{(iv)}(\alpha) + O(e_n^{13})
\end{aligned}
\tag{35}
$$

$$
\begin{aligned}
g'(y_n) &= g'(\alpha) + (y_n - \alpha)g''(\alpha) + \frac{(y_n - \alpha)^2}{2!}g'''(\alpha) + \frac{(y_n - \alpha)^3}{3!}g^{(iv)}(\alpha) \\
&+ \frac{(y_n - \alpha)^4}{4!}g^{(v)}(\alpha) + O(e_n^{13})
\end{aligned}
\tag{36}
$$

With the help of 3.17, 3.18, 3.19 and 3.20 we get

$$
z_n = \alpha + (-8c_2^2 c_3^3 + 24c_3^2 c_2^4 - 32c_3 c_2^6 + c_3^4 + 16c_2^8)e_n^9 + \ldots + O(e_n^{13})
\tag{37}
$$

$$
\begin{aligned}
f(z_n) &= f'(\alpha)[(z_n - \alpha) + c_2(z_n - \alpha)^2 + c_3(z_n - \alpha)^3 + c_4(z_n - \alpha)^4 \\
&+ O(e_n^{13})]
\end{aligned}
\tag{38}
$$

$$
\begin{aligned}
f'(z_n) &= f'(\alpha)[1 + 2c_2(z_n - \alpha) + 3c_3(z_n - \alpha)^2 + 4c_4(z_n - \alpha)^3 \\
&+ 5c_5(z_n - \alpha)^4 + O(e_n^{13})]
\end{aligned}
\tag{39}
$$

$$
\begin{aligned}
f''(z_n) &= f'(\alpha)[2c_2 + 6c_3(z_n - \alpha) + 12c_4(z_n - \alpha)^2 + 20c_5(z_n - \alpha)^3 \\
&+ 30c_6(z_n - \alpha)^4 + O(e_n^{13})]
\end{aligned}
\tag{40}
$$

Using equations $(3.12 - 3.22)$ in general iteration scheme(14), we get

$$
x_{n+1} = \alpha + \frac{80(-\frac{c_3}{2} + c_2^2)^4[(c_2^3 - c_2 c_3 - \frac{c_4}{5})g(\alpha) - \frac{2}{5}g'(\alpha)(-\frac{1}{2}c_3 + c_2^2)]}{g(\alpha)}e_n^{12} + O(e_n^{13}),
$$

which implies that

$$
e_{n+1} = \frac{80(-\frac{c_3}{2} + c_2^2)^4[(c_2^3 - c_2 c_3 - \frac{c_4}{5})g(\alpha) - \frac{2}{5}g'(\alpha)(-\frac{1}{2}c_3 + c_2^2)])}{g(\alpha)}e_n^{12} + O(e_n^{13}),
$$

which implies that the main and general iteration scheme(14) is of twelfth order of convergence and all algorithms deduced from it have also twelve order of convergence.

$\square$

4. Applications

In this section we included some nonlinear functions to illustrate the efficiency of our developed algorithms. We compare our developed algorithms with Newton's method (NM)[12] , Halley's method (HM) [7], Traub's Method (TM) [12] and modified Halley's method (MHM) [21] as follows:

Table 1: Comparison of various iterative methods				
Method	N	N_f	$|f(x_{n+1})|$	x_{n+1}
$f_1 = x^3 + 4x^2 - 10, x_0 = -0.3.$				
NM	54	108	$8.127500e - 30$	
HM	58	174	$6.815871e - 25$	
TM	27	81	$8.127500e - 30$	
MHM	22	66	$2.439119e - 36$	1.3652300134140968457608068
Algorithm 2.1	4	20	$3.107264e - 25$	
Algorithm 2.2	4	20	$4.374599e - 18$	
Algorithm 2.3	4	20	$3.366588e - 39$	

Table 2: Comparison of various iterative methods				
$f_2 = (x - 1)^3 - 2, x_0 = 0.1.$				
NM	13	26	$3.733284e - 27$	
HM	9	27	$6.698145e - 33$	
TM	7	21	$2.322902e - 54$	
MHM	7	21	$1.851813e - 15$	2.2599210498948731647672106
Algorithm 2.1	2	10	$1.321507e - 30$	
Algorithm 2.2	2	10	$6.843111e - 17$	
Algorithm 2.3	2	10	$7.873393e - 24$	

Table 3: Comparison of various iterative methods				
$f_3 = (x + 2)e^x - 1, x_0 = 3.1.$				
NM	9	18	$4.410873e - 15$	
HM	5	15	$3.594315e - 16$	
TM	5	15	$8.240223e - 30$	
MHM	5	15	$1.172300e - 56$	$-0.4428544010023885831413280$
Algorithm 2.1	2	10	$3.901801e - 15$	
Algorithm 2.2	3	15	$4.140026e - 17$	
Algorithm 2.3	3	15	$6.428238e - 36$	

Table 4: Comparison of various iterative methods				
$f_6 = e^{(x^2+7x-30)} - 1, x_0 = 3.3.$				
NM	9	18	$1.012535e - 20$	
HM	5	15	$1.168248e - 22$	
TM	5	15	$5.186798e - 41$	
MHM	7	21	$7.229220e - 21$	3.000000000000000000000000000
Algorithm 2.1	3	15	$2.975169e - 59$	
Algorithm 2.2	3	15	$2.517780e - 37$	
Algorithm 2.3	3	15	$3.078826e - 56$	

Table 5: Comparison of various iterative methods				
$f_4 = x^3 - x^2 + 3x \cos(x) - 1, x_0 = 1.2.$				
NM	10	20	$2.385992e - 16$	
HM	10	30	$6.209236e - 21$	
TM	5	15	$2.385992e - 16$	
MHM	9	27	$1.876664e - 28$	0.395323622986315188376142 8
Algorithm 2.1	4	20	$0.000000e + 00$	
Algorithm 2.2	3	15	$1.508088e - 15$	
Algorithm 2.3	3	15	$7.677213e - 16$	

Table 6: Comparison of various iterative methods				
$f_4 = \sin^2(x) - x^2 + 1, x_0 = 5.$				
NM	7	14	$1.502508e - 22$	
HM	4	12	$2.649470e - 38$	
TM	4	12	$7.125700e - 45$	
MHM	3	9	$4.868406e - 35$	1.404491648215341226035086 8
Algorithm 2.1	2	10	$4.427270e - 22$	
Algorithm 2.2	2	10	$5.124709e - 15$	
Algorithm 2.3	2	10	$7.738826e - 18$	

Table 7: Comparison of various iterative methods				
$f_5 = x + \ln(x), x_0 = 2.5.$				
NM	7	14	$2.441531e - 18$	
HM	4	12	$6.107599e - 22$	
TM	4	12	$1.213605e - 36$	
MHM	4	12	$3.863408e - 51$	0.5671432904097838729999687
Algorithm 2.1	2	10	$2.906122e - 39$	
Algorithm 2.2	2	10	$6.579576e - 41$	
Algorithm 2.3	2	10	$6.887373e - 39$	

Table 8: Comparison of various iterative methods				
$f_6 = e^x - \cos(x), x_0 = 2.7.$				
NM	8	16	$4.131186e - 21$	
HM	5	15	$5.752401e - 31$	
TM	4	12	$4.131186e - 21$	
MHM	4	12	$3.132281e - 27$	0.0000000000000000000000000
Algorithm 2.1	2	10	$4.961143e - 15$	
Algorithm 2.2	3	15	$2.628023e - 71$	
Algorithm 2.3	3	15	$1.186479e - 90$	

Table $(1-8)$**.** Shows the numerical comparisons of Newton's method, Halley's method, Traub's method, modified Halley's method and our developed methods. The columns represent the number of iterations N and the number of functions or derivatives evaluations N_f required to meet the stopping criteria, and the magnitude $|f(x)|$ of $f(x)$ at the final estimate x_n.

5. Conclusions

In recent years, researcher are keen in finding iterative methods for solving equation [22, 23, 24, 25, 26, 27]. This paper is an effect in this direction. In this paper, three new algorithms for solving non-linear functions has been established. We can concluded from tables $(1-7)$ that

(1) The efficiency index of our developed methods is $12^{\frac{1}{5}} \approx 1.6438$ which is greater than Newton's method, Halley's method and Noor and Noor's method.
(2) Our described methods are free from 3rd and higher derivatives, having twelfth order of convergence.
(3) By using some numerical examples the performance of our described methods is also discussed.Our methods are performing better as compared to Newton's method, Halley's method and Noor and Noor's method.

Competing interests

The authors declare that they have no competing interests.

References

[1] W. Nazeer, A. Naseem, S. M. Kang and Y. C. Kwun, *Generalized Newton Raphson's method free from second derivative,* J. Nonlinear Sci. Appl. 9 (2016), 2823-2831.

[2] W. Nazeer, M. Tanveer, S. M. Kang and A. Naseem., *A new Househölder's method free from second derivatives for solving nonlinear equations and polynomiography,* J. Nonlinear Sci. Appl. 9 (2016), 998-1007

[3] C. Chun, Construction of Newton-like iteration methods for solving nonlinear equations, Numeriche Mathematik 104 (2006) 297-315.

[4] R. L. Burden, J. D. Faires, Numerical Analysis (Sixth ed.), Brooks/Cole Publishing Company, Calif., 1997.

[5] J. Stoer, R. Bulirsch, Introduction to Numerical Analysis (Third ed.), Springer-Verlag, New York, 2002.

[6] A. Quarteroni, R. Sacco, F. Saleri, Numerical Mathematics, SpringerVerlag, New York, 2000.

[7] I. K. Argyros, A note on the Halley method in Banach spaces, Appl. Math. Comput. 58 (1993), pp. 215-224.

[8] J. M. Guti´errez, M. A. Hern´andez, An acceleration of Newtons method: super-Halley method, Appl. Math. Comput. 117 (2001), pp. 223-239.

[9] J. M. Guti´errez, M. A. Hern´andez, A family of Chebyshev-Halley type methods in Banach spaces, Bull. Austral. Math. Soc. 55 (1997), pp. 113-130.

[10] A. S. Househölder, The Numerical Treatment of a Single Nonlinear Equation, McGraw-Hill, New York, (1970).

[11] Newton's method and high order iterations Pascal Sebah and Xavier Gourdon numbers.computation.free.fr/Constants/constants.html October 3, 2001

[12] J. F. Traub, " Iterative Methods for the Solution of Equations ", Prentice-Hall, Englewood Cliffs, (NJ), 1964.

[13] M. Inokuti, H. Sekine and T. Mura, " General use of the Lagrange multiplier in nonlinear mathematical physics " in " Variational Methods in the Mechanics of Solids" (Ed. S. Nemat- Nasser), Pergamon Press, Oxford, 1978, pp.156-162.

[14] J. H. He, " Homotpy perturbation technique" Comput. Math. Appl. Mech. Eng., 1999, 178, 257- 262.

[15] J. H. He, " Variational iteration method-some recent results and new interpretations" J. Comp. Appl. Math., 2007, 207, 3-17.

[16] J. H. He, " Variational iteration method-a kind of nonlinear analytical technique: Some examples" Int. J. Nonlinear Mech., 1999, 34, 699-708.

[17] M. A. Noor, " New classes of iterative methods for nonlinear equations " Appl. Math. Comput., 2007, 191, 128-131.

[18] M. A. Noor and F. A. Shah, " Variational iteration technique for solving nonlinear equations "
biguplus, J. Appl. Math. Comput., 2009, 31, 247-254.

[19] J. Kuo, The improvements of modified Newton's method, Applied Mathematics and Computation, 189, 602-609.

[20] S. Abbasbandy, Improving Newton-Raphson method for nonlinear equations by modified Adomian decomposition method, Appl. Math. Comput., 145 (2003), 887-893.

[21] K. Inayat Noor, M. Aslam Noor, *Predictor-corrector Halley method for nonlinear equations,* Appl. Math. Comput, 188 (2007) 1587-1591

[22] Kang, S. M., Nazeer, W., Tanveer, M., Mehmood, Q., & Rehman, K. (2015). Improvements in Newton-Rapshon method for nonlinear equations using modified Adomian decomposition method. International Journal of Mathematical Analysis, 9(39), 1919-1928.

[23] Kang, S. M., Nazeer, W., Rafiq, A., & Youg, C. Y. (2014). A new third order iterative method for scalar nonlinear equations. International Journal of Mathematical Analysis, 8(43), 2141-2150.

[24] Nazeer, W., Kang, S. M., & Tanveer, M. (2015). Modified Abbasbandy's Method for Solving Nonlinear Functions with Convergence of Order Six. International Journal of Mathematical Analysis, 9(41), 2011-2019.

[25] Kwun, Y. C., Fahad, M., Nazeer, W., Tanveer, M., & Kang, S. M. (2017). Fixed point type iterative method for solving nonlinear equations and polynomiography. Far East Journal of Mathematical Sciences, 102(2), 251-269.

[26] Kang, S. M., Ramay, S. M., Tanveer, M., & Nazeer, W. (2016). Polynomiography via an iterative method corresponding to Simpson's 1/3 rule. JOURNAL OF NONLINEAR SCIENCES AND APPLICATIONS, 9(3), 967-976.

[27] Kwun, Y. C., Munir, M., Nazeer, W., & Kang, S. M. (2016). Some fixed points results of quadratic functions in split quaternions. Journal of Function Spaces, 2016.