

New algorithms for nonlinear equations



Muhammad Idrees¹, Hongbin Ma^{1,*}, Amir Naseem², Sajid Ali³, Abdul Rauf Nizami⁴

¹School of Automation, Beijing Institute of Technology, Beijing, China

²Department of Mathematics, Lahore Leads University, Lahore, Pakistan

³Department of Computer Science, University of Education, DG Khan, Pakistan

⁴Abdus Salam School of Mathematical Sciences, GC University, Lahore, Pakistan

ARTICLE INFO

Article history:

Received 11 March 2018

Received in revised form

20 June 2018

Accepted 10 July 2018

Keywords:

Nonlinear equation

Newton's method

Halley's method

Householder's method

ABSTRACT

In this paper, three new algorithms are introduced to solve non-linear equations using variational iteration technique, and their convergence criteria is discussed. Moreover, several examples are also given to illustrate that these algorithms are more efficient than Newton's method, Halley's method, and Householder's method. Besides, one can confirm the efficiency of the introduced fast algorithms in terms of the so-called efficiency index. Such techniques have potential applications in adaptive estimation and control, parameter estimation, and so on.

© 2018 The Authors. Published by IASE. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

Many problems in mathematics, physics, and engineering sciences are related to solutions of non-linear equations. For example, in statistics or adaptive estimation and control, how to estimate unknown parameter is one critical part which can be usually converted to a problem of solving non-linear equations. In practice, more and more unmanned systems such as unmanned aerial vehicle (UAV) and unmanned ground vehicle (UGV) have witnessed interests of researchers and their dynamics equations are often nonlinear in real world. Up to now, a majority of methods for coping with nonlinear systems or equations are still based on linearization. However, sometimes the considered systems are highly nonlinear. When linearization is used to solve these problems, accumulative error can be very large. If we can find solutions of these nonlinear equations directly instead of linearization, the accuracy of unmanned systems can be improved. Since most of the non-linear equations do not have exact solutions, we are obliged to find their numerical solutions, which are iterative in nature.

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 (Nazeer et al., 2016a; 2016b; Chun, 2006; Burden and Faires, 1997;

Stoer and Bulirsch, 2002; Quarteroni et al., 2000; Chen et al., 1993; Householder, 1970; Traub, 1982; Inokuti et al., 1978; He, 1999a; 1999b; 2007; Noor, 2007; Shah et al., 2016; Thukral, 2015; 2016). The most famous method for solving non-linear equations is Newton's method (Nazeer et al., 2016a). To improve its convergence, various modified methods have been developed in literature (Chun, 2006; Gutierrez and Hernandez, 1997; 2001; Householder, 1970; Sebah and Gourdon, 2001; Abbasbandy, 2003). Chun (2006) generalized and modified generalized Newton-Raphson methods are suggested, which are free from second derivatives. In Gutierrez and Hernandez (1997), the convergence analysis for the Halley's method is simplified and the best possible error bound under standard Newton-Kantorovich hypotheses has been obtained. In Sebah and Gourdon (2001), global convergence theorems for Halley's and chebyshev's methods have been proved. In Noor and Noor (2007), a new two-step iterative method for solving nonlinear equations has been given, known as the predictor-corrector Halley's method with convergence of order 6 which was considered a significant improvement as compared to previously known methods.

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 a predictor function having convergence order q where $q \geq 1$ and the second function $\psi(x)$ acts as a 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

* Corresponding Author.

Email Address: mathmh@bit.edu.cn (H. Ma)

<https://doi.org/10.21833/ijaas.2018.09.006>

2313-626X/© 2018 The Authors. Published by IASE.

This is an open access article under the CC BY-NC-ND license

(<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

variational iteration technique was introduced by Inokuti et al. (1978). Using this technique, Noor and Shah (2009) and Noor (2007) derived some iterative methods for solving the non-linear equations. The purpose of this technique was to solve variety of diverse problems (He, 1999a; 1999b; 2007). We apply this technique to obtain three higher-order iterative methods. These methods have convergence order twelve, which is remarkable. Finally, several examples are given to illustrate their performance.

2. Construction of iterative methods

Consider a nonlinear equation

$$f(x) = 0 \tag{1}$$

with α as a simple root and γ an initial guess, sufficiently close to α . For the sake of convenience, we consider an approximate solution x_n such that $f(x_n) \neq 0$. Furthermore, consider $\phi(x)$ and $\psi(x)$ as iteration functions of order q and r , respectively. Then

$$x_{n+1} = \phi(x_n) + \mu [f(\psi(x_n))g(\psi(x_n))]^t, \tag{2}$$

where $t = \frac{q}{r}$ is a recurrence relation which generates iterative methods of order $q + r$, $g(x)$ is any arbitrary function which will be converted later on 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, we have

$$\mu = - \frac{\phi'(x_n)[f(\psi(x_n))g(\psi(x_n))]^{1-t}}{t\psi'(x_n)[f'(\psi(x_n))g(\psi(x_n))+f(\psi(x_n))g'(\psi(x_n))]} \tag{3}$$

From Eq. 2 and Eq. 3, we get

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

Now we apply Eq. 4 for constructing a general iterative scheme: Suppose that

$$\psi(x_n) = y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}, \tag{5}$$

which is well known Householder’s method with cubic convergence. With the help of Eq. 4 and Eq. 5, we can write

$$x_{n+1} = \phi(x_n) - \frac{\phi'(x_n)f(y_n)g(y_n)}{t\psi'(x_n)[f'(y_n)g(y_n)+f(y_n)g'(y_n)]} \tag{6}$$

$$\phi(x_n) = z_n = y_n - \frac{f(y_n)}{f'(y_n)} - \frac{f^2(y_n)f''(y_n)}{2f'^3(y_n)}, \tag{7}$$

where $y_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}$. This is two-step Householder method with convergence order nine. Differentiating $\phi(x_n)$ with respect to x we get

$$\phi'(x_n) = z'_n = \frac{f^2(y_n)[3f''^2(y_n)-f'(y_n)f'''(y_n)]}{2f'^4(y_n)} y'_n \tag{8}$$

and from Taylor’s series of $f''(z_n)$, we get

$$f''(z_n) = f''(y_n) + (y_n - z_n)f'''(y_n) + \dots \tag{9}$$

Ignoring squared and higher powers of $(y_n - z_n)$ yields

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

with the help of Eq. 8 and Eq. 10, we get

$$\phi'(x_n) = \frac{f(y_n)Q_n}{2f'^4(y_n)[2f''^2(y_n)+f'(y_n)f'''(y_n)]} y'_n \tag{11}$$

where

$$Q_n = 2f'^4(y_n)f''(z_n) - 2f'^4(y_n)f''(y_n) + 6f'(y_n)f''^2(y_n)f''^2(y_n) + 3f''^3(y_n)f^2(y_n) \tag{12}$$

substituting Eq. 10 in Eq. 6, we obtain

$$x_{n+1} = z_n - \frac{f(y_n)g(y_n)\phi'(x_n)}{t[f'(y_n)g(y_n)+f(y_n)g'(y_n)]} \tag{13}$$

Since $q = 9$ and $r = 3$, then $t = \frac{q}{r} = \frac{9}{3} = 3$. Then Eq. 13 becomes

$$x_{n+1} = z_n - \frac{f(y_n)g(y_n)\phi'(x_n)}{3[f'(y_n)g(y_n)+f(y_n)g'(y_n)]} \tag{14}$$

The relation in Eq. 14 is the main and general iterative scheme, which we use to deduce iterative methods for solving nonlinear equations by considering some special cases of the auxiliary functions g .

2.1. Algorithm I

Let $g(x_n) = \exp(-\beta x_n)$. Then $g'(x_n) = -\beta g(x_n)$. Now using Eq. 14, we obtain the algorithm:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}, \\ z_n &= y_n - \frac{f(y_n)}{f'(y_n)} - \frac{f^2(y_n)f''(y_n)}{2f'^3(y_n)} \\ x_{n+1} &= z_n - \frac{f(y_n)\phi'(x_n)}{3[f'(y_n)-\beta f(y_n)]}, \text{ for } n = 0,1,2,\dots \end{aligned}$$

2.2. Algorithm II

Let $g(x_n) = \exp(-\beta f(x_n))$. Then $g'(x_n) = -\beta f'(x_n)g(x_n)$. Now using Eq. 14, we obtain the algorithm:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}, \\ z_n &= y_n - \frac{f(y_n)}{f'(y_n)} - \frac{f^2(y_n)f''(y_n)}{2f'^3(y_n)} \\ x_{n+1} &= z_n - \frac{f(y_n)\phi'(x_n)}{3[f'(y_n)-\beta f'(y_n)f'(y_n)]}, \text{ for } n = 0,1,2, \dots \end{aligned}$$

2.3. Algorithm III

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 Eq. 14, we obtain the following algorithm

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

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

$$x_{n+1} = z_n - \frac{f(y_n)f''(y_n)\phi'(x_n)}{3[f'^3(y_n)+\beta f(y_n)f''(y_n)]}, \text{ for } n = 0,1,2,\dots$$

Remark 2.1: These algorithms, whose convergence is proved in the next section, are suitable for large values of β .

3. Convergence analysis

In this section, we discuss the convergence order of the general iteration scheme in [Inokuti et al. \(1978\)](#).

Theorem 3.1: Suppose α is a root of the equation $f(x) = 0$, and $f(x)$ is sufficiently smooth near α . Then the convergence order of Eq. 14 is at least twelve.

Proof: Suppose that α is a root of the equation $f(x) = 0$ and e_n is the error at n th iteration, i.e. $e_n = x_n - \alpha$. Using Taylor series expansion, we get

$$f(x) = f'(\alpha)e_n + \frac{1}{2!}f''(\alpha)e_n^2 + \frac{1}{3!}f'''(\alpha)e_n^3 + \frac{1}{4!}f^{(iv)}(\alpha)e_n^4 + \frac{1}{5!}f^{(v)}(\alpha)e_n^5 + \frac{1}{6!}f^{(vi)}(\alpha)e_n^6 + \dots + O(e_n^{13}) \tag{15}$$

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

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

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

here $c_n = \frac{1}{n!} \frac{f^{(n)}(\alpha)}{f'(\alpha)}$. Now using Eq. 15, Eq. 16 and Eq. 17, we get

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

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

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

$$f''(y_n) = f''(\alpha)[2c_2 + (-6c_2^3 + 12c_3c_2^2)e_n^3 + (72c_2c_2^2 - 54c_3c_3^2 - 18c_4c_3)e_n^4 + (90c_3^3 - 378c_2^2c_3^2 + 180c_3c_4^2 + 144c_2c_4c_3 - 36c_3c_5)e_n^5 + (-60c_3c_6 + 240c_3c_2c_5 + 342c_4c_3^2 - 720c_3c_4c_2^2 - 816c_2c_3^3 + 1506c_2^3c_3^2 - 528c_3c_2^5 + 48c_4c_2^4)e_n^6 + \dots + O(e_n^{13})]$$

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

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

it follows that

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

$$f(z_n) = f'(\alpha)[(-8c_2^2c_3^3 + 24c_3^2c_2^4 - 32c_3c_2^6 + c_3^4 + 16c_2^8)e_n^9 + \dots + O(e_n^{13})]$$

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

$$f''(z_n) = f''(\alpha)[2c_2 + (-48c_2^2c_3^3 + 144c_3^3c_2^4 - 192c_3^2c_2^6 + 6c_3^5 + 96c_3c_2^8)e_n^9 + \dots + O(e_n^{13})]$$

using these relations in Eq. 14, we have

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

which implies that

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

and the proof is finished.

4. Numerical examples

In this section, comparison of different iterative methods is given. The following examples given in [Tables 1-8](#) confirm that the algorithms I, II, and III are more efficient than the Newton's method (NM) ([Nazeer et al., 2016a; 2016b](#)), Halley's method (HM) ([Chen et al., 1993](#)), Traub's Method (TM) ([Traub, 1982](#)), and modified Halley's method (MHM) ([Noor, 2007](#)).

Table 1: Results of example 1

$f_1 = x^3 + 4x^2 - 10, x_0 = -0.3.$				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
NM	54	108	$8.127500 * 10^{-30}$	
HM	58	174	$6.815871 * 10^{-25}$	
TM	27	81	$8.127500 * 10^{-30}$	
MHM	22	66	$2.439119 * 10^{-36}$	1.3652300134
Alg I	4	20	$3.107264 * 10^{-25}$	
Alg II	4	20	$4.374599 * 10^{-18}$	
Alg III	4	20	$3.366588 * 10^{-39}$	

Table 2: Results of example 2

$f_2 = (x - 1)^3 - 2, x_0 = 0.1.$				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
NM	13	26	$3.733284 * 10^{-27}$	
HM	9	27	$6.698145 * 10^{-33}$	
TM	7	21	$2.322902 * 10^{-54}$	
MHM	7	21	$1.851813 * 10^{-15}$	2.2599210498
Alg I	2	10	$1.321507 * 10^{-30}$	
Alg II	2	10	$6.843111 * 10^{-17}$	
Alg III	2	10	$7.873393 * 10^{-24}$	

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 . The term “efficiency index” (Nazeer et al., 2016a; 2016b) shows us how fast and efficient our method is. It is used to analyze the performance of different iterative methods. The efficiency index of our method is $12^{\frac{1}{5}} \approx 1.6438$.

Table 3: Results of example 3

$f_3 = (x+2)e^x - 1, x_0 = 3.1.$				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
NM	9	18	$4.410873 * 10^{-15}$	-0.4428544010
HM	5	15	$3.594315 * 10^{-16}$	
TM	5	15	$8.240223 * 10^{-30}$	
MHM	5	15	$1.172300 * 10^{-56}$	
Alg I	2	10	$3.901801 * 10^{-15}$	
Alg II	3	15	$4.140026 * 10^{-17}$	
Alg III	3	15	$6.428238 * 10^{-36}$	

Table 4: Results of example 4

$f_4 = e^{x^2-7x-30}, x_0 = 3.3$				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
NM	9	18	$1.012535 * 10^{-20}$	3.0000000000
HM	5	15	$1.168248 * 10^{-22}$	
TM	5	15	$5.186798 * 10^{-41}$	
MHM	7	21	$7.229220 * 10^{-21}$	
Alg I	3	15	$2.975169 * 10^{-59}$	
Alg II	3	15	$2.517780 * 10^{-37}$	
Alg III	3	15	$3.078826 * 10^{-56}$	

Table 5: Results of example 5

$f_5 = x^3 - x^2 + 3x\cos(x) - 1, x_0 = 1.2.$				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
NM	10	20	$2.385992 * 10^{-16}$	0.3953236229
HM	10	30	$6.209236 * 10^{-21}$	
TM	5	15	$2.385992 * 10^{-16}$	
MHM	9	27	$1.876664 * 10^{-28}$	
Alg I	4	20	$0.000000 * 10^{+00}$	
Alg II	3	15	$1.508088 * 10^{-15}$	
Alg III	3	15	$7.677213 * 10^{-16}$	

Table 6: Results of example 6

$f_6 = \sin^2(x) - x^2 + 1, x_0 = 5.$				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
NM	7	14	$1.502508 * 10^{-22}$	1.4044916482
HM	4	12	$2.649470 * 10^{-38}$	
TM	4	12	$7.125700 * 10^{-45}$	
MHM	3	9	$4.868406 * 10^{-35}$	
Alg I	2	10	$4.427270 * 10^{-22}$	
Alg II	2	10	$5.124709 * 10^{-15}$	
Alg III	2	10	$7.738826 * 10^{-18}$	

Table 7: Results of example 7

$f_7 = x + \ln(x), x_0 = 2.5.$				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
NM	N_f	14	$2.441531 * 10^{-18}$	0.5671432904
HM	4	12	$6.107599 * 10^{-22}$	
TM	4	12	$1.213605 * 10^{-36}$	
MHM	4	12	$3.863408 * 10^{-51}$	
Alg I	2	10	$2.906122 * 10^{-39}$	
Alg II	2	10	$6.579576 * 10^{-41}$	
Alg III	2	10	$6.887373 * 10^{-39}$	

Table 8: Results of example 8

$f_8 = e^x - \cos(x), x_0 = 2.7.$				
Method	N	N_f	$ f(x_{n+1}) $	x_{n+1}
NM	8	16	$4.131186 * 10^{-21}$	0.0000000000
HM	5	15	$5.752401 * 10^{-31}$	
TM	4	12	$4.131186 * 10^{-21}$	
MHM	4	12	$3.132281 * 10^{-27}$	
Alg I	2	10	$4.961143 * 10^{-15}$	
Alg II	3	15	$2.628023 * 10^{-71}$	
Alg III	3	15	$1.186479 * 10^{-90}$	

5. Conclusion

The introduction of three new algorithms aims to several simple yet powerful mathematical tips for improving the efficiency of solving nonlinear equations. Their efficiency index is greater than Newton’s method (Nazeer et al., 2016a; 2016b), Halley’s method (Chen et al., 1993), Traub’s Method (Traub, 1982) and Noor’s method (Noor, 2007). In our future work, we will apply our methods for more complex examples. We will also present polynomiography of our methods. An adaptive control problem will be solved with a better estimation of parameter.

Authors’ contributions

Muhammad Idrees developed the main idea of the study, participated in the sequence alignment and drafted the manuscript. Ma Hongbin refined the idea and supervised the study, participated in its design, coordination, and helped to draft the manuscript. Amir Naseem, Abdur Rauf Nizami, and Sajid Ali and Li Song helped in analysis and software operations. All authors have read and approved the final manuscript.

Acknowledgement

This work is partially supported by the National Key Research and Development Program of China under Grant 2017YFF0205306 and National Natural Science Foundation (NSFC) under Grants 61473038 and 91648117. And this work is also partially supported by Beijing Natural Science Foundation (BJNSF) under Grant 4172055.

References

Abbasbandy S (2003). Improving Newton–Raphson method for nonlinear equations by modified Adomian decomposition method. *Applied Mathematics and Computation*, 145(2-3): 887-893.

Burden RL and Faires JD (1997). *Numerical analysis*. 6th Edition, Cole Publishing Company, Three Lakes, Wisconsin, USA.

Chen D, Argyros IK, and Qian QS (1993). A note on the Halley method in Banach spaces. *Applied Mathematics and Computation*, 58(2-3): 215-224.

Chun C (2006). Construction of Newton-like iteration methods for solving nonlinear equations. *Numerische Mathematik*, 104(3): 297-315.

Gutierrez JM and Hernandez MA (1997). A family of Chebyshev-Halley type methods in Banach spaces. *Bulletin of the Australian Mathematical Society*, 55(1): 113-130.

Gutierrez JM and Hernandez MA (2001). An acceleration of Newton’s method: Super-Halley method. *Applied Mathematics and Computation*, 117(2-3): 223-239.

He JH (1999a). Homotopy perturbation method. *Computer Methods in Applied Mechanics and Engineering*, 178(3-4): 257-262.

He JH (1999b). Variational iteration method—a kind of non-linear analytical technique: Some examples. *International Journal of Non-Linear Mechanics*, 34(4): 699-708.

- He JH (2007). Variational iteration method—some recent results and new interpretations. *Journal of Computational and Applied Mathematics*, 207(1): 3-17.
- Householder AS (1970). *The numerical treatment of a single nonlinear equation*. McGraw-Hill, New York, USA.
- Inokuti M, Sekine H, and Mura T (1978). General use of the lagrange multiplier in nonlinear mathematical physics. *Variational Method in the Mechanics of Solids*, 33(5): 156-162.
- Nazeer W, Naseem A, Kang SM, and Kwun YC (2016a). Generalized Newton Raphson's method free from second derivative. *Journal Nonlinear Science and Applications*, 9: 2823-2831.
- Nazeer W, Tanveer M, Kang SM, and Naseem A (2016b). A new Householder's method free from second derivatives for solving nonlinear equations and polynomiography. *Journal Nonlinear Science and Applications*, 9: 998-1007.
- Noor KI and Noor MA (2007). Predictor-corrector Halley method for nonlinear equations. *Applied Mathematics and Computation*, 188(2): 1587–1591.
- Noor MA (2007). New classes of iterative methods for nonlinear equations. *Applied Mathematics and Computation*, 191(1): 128-131.
- Noor MA and Shah FA (2009). Variational iteration technique for solving nonlinear equations. *Journal of Applied Mathematics and Computing*, 31(1-2): 247-254.
- Quarteroni A, Sacco R, and Saleri F (2000). *Numerical Mathematics*. Springer, Berlin, Germany.
- Sebah P and Gourdon X (2001). Newton's method and high order iterations, Technical Report. Available online at: <http://numbers.computation.free.fr/Constants/Algorithms/newton.html>
- Shah FA, Noor MA, and Noor KI (2016). Some iterative schemes for obtaining approximate solution of nonlinear equations. *University Politehnica of Bucharest Scientific Bulletin-Series A-Applied Mathematics and Physics*, 78(1): 59-70.
- Stoer J and Bulirsch R (2002). *Introduction to numerical analysis*. 3rd Edition, Springer-Verlag, New York, USA.
- Thukral R (2015). New modifications of Newton-type methods with eighth order convergence for solving nonlinear equations. *Journal of Advances in Mathematics*, 10(3): 3362-3373.
- Thukral RA (2016). Newton-type method with ninth-order convergence for solving nonlinear equations. *Journal of Numerical Mathematics and Stochastics*, 8(1): 9-16.
- Traub JF (1982). *Iterative methods for the solution of equations* (Vol. 312). American Mathematical Society, Providence, Rhode Island, USA.