Mathematical Science Letters
An International Journal

http://dx.doi.org/10.12785/msl/020205

Higher-order iterative methods by using Householder's method for solving certain nonlinear equations

Waseem Asghar Khan[1], Muhammad Aslam Noor [2] and Adnan Rauf[3]

[1]Department of Mathematics, Sukkar-Institute of Business Administration, Sukkur-65200 Sindh, Pakistan
[2]Department of Mathematics, COMSATS Institute of Information Technology, Islamabad, Pakistan
[3]Department of Mathematics, Institute of Business Management, Karongi Creek Karachi, Pakistan

Email: waseemasg@gmail.com, noormaslam@hotmail.com, adnan.rauf@gmail.com

Received: 5 Jan 2013; Revised: 7 Apr 2013; Accepted: 8 Apr 2013
Published online: 1 May. 2013

Abstract: In this paper, we suggest and analyze some new higher-order iterative methods by using Householder's method free from second derivative for solving nonlinear equations. Here we use new and different technique for implementation of higher-order derivative of the function and derive new higher-order predictor-corrector iterative methods free from second derivative. The efficiency index equals to $9^{1/5} \approx 1.552$. Several numerical examples are given to illustrate the efficiency and performance of these new methods.

Keywords: Nonlinear equations; Newton method; Convergence criteria; Root finding method; Numerical examples.

1 Introduction

It is well known that a wide class of problem which arises in several branches of pure and applied science can be studied in the general framework of the nonlinear equations $f(x)=0,$ Due to their importance; several numerical methods have been suggested and analyzed under certain conditions. These numerical methods have been constructed using different techniques such as Taylor series, homotopy perturbation method [22-23] and its variant forms, quadrature formula, variational iteration method, and decomposition method; see, for example [1-23]. Using the technique of updating the solution and Taylor series expansion, Noor and Noor [14] have suggested and analyzed a sixth-order predictor-corrector iterative type Halley method for solving the nonlinear equations. Ham et al. [7] and Chun [4] have also suggested a class of fifth-order and sixth-order iterative methods. In the implementation of the method [14], one has to evaluate the second derivative of the function, which is a serious drawback of these methods. To overcome these drawbacks, we modify the predictor-corrector Halley method by replacing the second derivatives of the function by its suitable scheme. We prove that the new modified predictor-corrector method is of sixth-order convergence free from second derivatives. We also present the comparison of the new method with the methods of Ham et al. [7] and Chun [4]. Several examples are given to illustrate the efficiency and robustness of the new proposed method.

It has been shown that these new iterative methods include a wide class of known and new iterative methods as special cases. Also discuss the efficiency index and computational order of convergence of new methods. Several examples are given to illustrate the efficiency and performance of these new methods. We also compare these new methods with other recent methods of the same convergence order.

2 Iterative methods

We recall the Newton's method [6] and Householder's method [3,5] in Algorithm 2.1 and Algorithm 2.2, we have

Algorithm 2.1 For a given x_0, compute approximates solution x_{n+1} by the iterative scheme:

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

Algorithm 2.1 is the well-known Newton method, which has a quadratic convergence [6].

Algorithm 2.2 For a given x_0, compute approximates solution x_{n+1} by the iterative schemes:

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

This is known as Householder's method, which has cubic convergence [3, 5].
Noor and Noor [13], have suggested the following two-step method, using Algorithm 2.1 method as predictor and Algorithm 2.2 as a corrector.

Algorithm 2.3 For a given x_0, compute approximates solution x_{n+1} by the iterative schemes:

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

If $f''(y_n) = 0$, then Algorithm 2.3 is called the predictor-corrector method and has fourth-order convergence, see [6]. In order to implement this method, one has to find the second derivative of this function, which may create some problems. To overcome this drawback, we use new and different technique to reduce second derivative of the function into the first derivative. This idea plays a significant role in developing some new iterative methods free from second derivatives. To be more precise, we consider

$$f''(y_n) = \frac{2}{y_n - x_n}\left\{2f'(y_n) + f'(x_n) - 3\frac{f(y_n) - f(x_n)}{y_n - x_n}\right\} \equiv P_f(x_n, y_n). \tag{4}$$

Combining (3) and (4), we suggest the following new iterative method for solving the nonlinear equation (1) and this is the new motivation of higher-order.

Algorithm 2.4 For a given x_0, compute approximates solution x_{n+1} by the iterative schemes:

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

Algorithm 2.4 is called the new two-step modified Householder's method free from second derivative for solving nonlinear equation (1). This method has sixth-order convergence. Per iteration this method requires two evaluations of the function and two evaluations of its first-derivative, so its efficiency index equals to $6^{1/4} \approx 1.565$, if we consider the definition of efficiency index [18] as $p^{1/m}$, where p is the order of the method and m is the number of functional evaluations per iteration required by the method.

Following the technique of predictor-corrector of the solution, see [4, 7]. We derive the new methods, we have

Algorithm 2.5: For a given x_0, compute approximates solution x_{n+1} by the iterative schemes:

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

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

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

Algorithm 2.6: For a given x_0, compute approximates solution x_{n+1} by the iterative schemes:

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

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

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

These new methods have seventh-order convergence. Per iteration this method requires two evaluations of the function and two evaluations of its first-derivative, so its efficiency index equals to $7^{1/4} \approx 1.475.$

Algorithm 2.7: For a given x_0, compute approximates solution x_{n+1} by the iterative schemes:

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

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

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

Algorithm 2.8: For a given x_0, compute approximates solution x_{n+1} by the iterative schemes:

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

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

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

These new methods in Algorithm 2.7 and Algorithm 2.8 have eighth-order convergence. Per iteration these methods requires two evaluations of the function and two evaluations of its first-derivative, so its efficiency index equals to $8^{1/4} \approx 1.515$.

In the similar way, we can suggest the following new iterative methods.

Algorithm 2.9: For a given x_0, compute approximates solution x_{n+1} by the iterative schemes:

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

$$z_n = y_n - \frac{f(y_n)}{f'(y_n)} - \frac{f(y_n)P_f(x_n, y_n)}{2f'^3(y_n)}. \quad (5)$$

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

Algorithm 2.10: For a given x_0, compute approximates solution x_{n+1} by the iterative schemes:

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

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

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

3 Convergence criteria

Now we consider the convergence criteria of Algorithm 2.9. In a similar way, we can discuss the convergence of other Algorithms.

Theorem 1: Let $\alpha \in D$ be a simple zero of sufficiently differentiable function $f : D \subset R \to R$ for an open interval D. And x_0 is initial choice, then Algorithm 2.9 has ninth-order convergences.

Proof. If α is the root and e_n be the error at *nth* iteration, than $e_n = x_n - \alpha$, using Taylor's expansion, we have

$$f(x_n) = f'(x_n)e_n + \frac{1}{2!} f''(x_n)e_n^{\ 2} + \frac{1}{3!} f'''(x_n)e_n^{\ 3} + \frac{1}{4!} f^{(iv)}(x_n)e_n^{\ 4} + \frac{1}{5!} f^{(v)}(x_n)e_n^{\ 5}$$
$$+ \frac{1}{6!} f^{(vi)}(x_n)e_n^{\ 6} + O(e_n^{\ 7}),$$

$$f(x_n) = f'(\alpha)[e_n + c_2 e_n^{\ 2} + c_3 e_n^{\ 3} + c_4 e_n^{\ 4} + c_5 e_n^{\ 5} + O(e_n^{\ 6})], \quad (7)$$

$$f'(x_n) = f'(\alpha)[1+2c_2e_n+3c_3e_n{}^2+4c_4e_n{}^3+5c_5e_n{}^4+6c_6e_n{}^5+O(e_n{}^6)], \tag{8}$$

where

$$c_k = \frac{1}{k!}\frac{f^{(k)}(\alpha)}{f'(\alpha)}, \qquad k=2,3,\ldots, \quad \text{let } \ e_n = x_n-\alpha, \ \text{and}$$

From (7) and (8), we have

$$\frac{f(x_n)}{f'(x_n)} = e_n - c_2e_n{}^2-(2c_3-2c_2^2)e_n{}^3-(3c_4-7c_2c_3+4c_2^3)e_n{}^4+(-6c_3{}^2+20c_3c_2^2$$
$$-10c_2c_4+4c_5-8c_2^4)e_n{}^5+O(e_n{}^6). \tag{9}$$

From equation (9), we have

$$y_n = \alpha + c_2e_n{}^2+(2c_3-2c_2^2)e_n{}^3+(3c_4-7c_2c_3+4c_2^3)e_n{}^4+O(e_n{}^5). \tag{10}$$

$$f(y_n) = f'(\alpha)[c_2e_n{}^2+2(c_3-c_2^2)e_n{}^3+(3c_4-7c_2c_3+5c_2^3)e_n{}^4+O(e_n{}^5)], \tag{11}$$

and,

$$f'(y_n) = f'(\alpha)[1+c_2^2e_n{}^2+4(c_2c_3-c_2^3)e_n{}^3+(8c_2^4+6c_2c_4-11c_2^2c_3)e_n{}^4+O(e_n{}^5). \tag{12}$$

$$\frac{f(y_n)P_f(x_n,y_n)}{2f'^2(y_n)} = c_2^2e_n{}^2+(2c_2c_3-2c_2^3)e_n{}^3+(c_2^4+2c_2c_4-4c_2^2c_3)e_n{}^4+(-4c_3c_2^3$$
$$+4c_2^5+2c_2c_5-6c_2^2c_4+6c_2c_3^2-2c_3c_4)e_n{}^5+(36c_3c_2^4+6c_4c_2^3 \tag{13}$$
$$-12c_2^6-45c_3^2c_2^2-8c_2^2c_5-3c_4^2-4c_5c_3+2c_2c_6+16c_2c_3c_4+6c_3^3)e_n{}^6$$
$$+O(e_n{}^7).$$

$$z_{n+1} = y_n - f(y_n)\Big/ f'(y_n) - \frac{f(y_n)P_f(x_n,y_n)}{2f'^3(y_n)} = (c_2^2c_4-c_3c_2^3+c_2^5)e_n{}^6$$
$$+(2c_5c_2^2+4c_2c_3c_4-6c_4c_2^3-6c_3^2c_2^2+12c_3c_2^4-6c_2^6)e_n{}^7+(21c_2^7+3c_6c_2^2 \tag{14}$$
$$-43c_2^2c_3c_4+288c_2c_3c_5-9c_2^3c_5-63c_3c_2^5-12c_2c_3^3+29c_4c_2^4$$
$$+6c_2c_4^2+4c_4c_3^2+57c_3^2c_2^3)e_n{}^8+O(e_n{}^9).$$

Using (7)-(14) in Algorithm 2.9, we have

$$x_{n+1} = \alpha + (2c_2^8+2c_4c_2^5-4c_3c_2^6+2c_3^2c_2^4-2c_3c_4c_2^3)e_n^9+O(e_n^{10}).$$

Thus, we have

$$e_{n+1} = (2c_2^8+2c_4c_2^5-4c_3c_2^6+2c_3^2c_2^4-2c_3c_4c_2^3)e_n^9+O(e_n^{10}).$$

which shows that Algorithm 2.9 has ninth-order convergence.

4 Numerical examples

In this section, we present some numerical examples to illustrate the efficiency and the accuracy of the new developed iterative methods in this paper (Table 1-Table 7). We compare our new methods obtained in Algorithm 2.4 to Algorithm 2.10 with Newton's method (NM), method of Noor and Noor ([14], NN1), method of Noor et al. ([16], NK), methods of Chun ([7], CM1, CM2 and CM3), method of Siyyam ([19], SM), method of Li and Jiao ([9], LJ) and method of Javidi ([8], JM1 and JM2). All computations have been done by using the Maple 11 package with 25 digit floating point arithmetic. We accept an approximate solution rather than the exact root, depending on the precision (ε) of the computer. We use the following stopping criteria for computer programs:

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

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

and so, when the stopping criterion is satisfied, x_{n+1} is taken as the exact root α computed. For numerical illustrations we have used the fixed stopping criterion $\varepsilon = 10^{-15}$. As for the convergence criteria, it was required that the distance of two consecutive approximations δ. Also displayed are the number of iterations to approximate the zero (IT), the approximate root x_n, the value $f(x_n)$ and the computational order of convergence (COC) can be approximated using the formula, $COC \approx \dfrac{\ln|(x_{n+1}-x_n)/(x_n - x_{n-1})|}{\ln|(x_n - x_{n-1})/(x_{n-1}-x_{n-2})|}$. All examples are same in [3].

Example 1. Consider the equation $f_1(x) = x^3 + 4x^2 - 10, \quad x_0 = 1.$

Table 1 (Approximate solution of example 1)

Methods	*IT*	x_n	$f(x_n)$	δ	*COC*
NM	6	1.365230013414096845760806 8	3.98235e-43	2.21790e-22	2
NN1	3	1.3652300134140968457608068	0	5.58014e-26	6.11
NK	3	1.3652300134140968457608068	0	2.78777e-19	5.16
Alg. 2.4	3	1.3652300134140968457608068	0	5.58014e-26	6.11
Alg. 2.5	3	1.3652300134140968457608068	0	2.93030e-37	7.11
Alg. 2.6	3	1.3652300134140968457608068	0	2.12459e-27	7.26
Alg. 2.7	3	1.3652300134140968457608068	0	7.35071e-43	8.11
Alg. 2.8	3	1.3652300134140968457608068	0	7.12655e-43	8.11
Alg. 2.9	3	1.3652300134140968457608068	0	1.81330e-55	9.11
Alg. 2.10	3	1.3652300134140968457608068	0	4.52450e-50	9.16
JM1	4	1.3652300134140968457608068	0	2.41159e-44	4

JM2	4	1.3652300134140968457608068	0	0	-
LJM	4	1.3652300134140968457608068	0	9.46019e-37	5.02
SM	4	1.3652300134140968457608068	0	1.73961e-50	5
CM1	4	1.3652300134140968457608068	0	0	-
CM2	3	1.3652300134140968457608068	0	7.07294e-23	6.08
CM3	3	1.3652300134140968457608068	0	2.08447e-20	5.03

Example 2 Consider the equation $f_2(x)=\sin^2 x-x^2+1, \quad x_0=1.3.$

Table.2 (Approximate solution of example 2)

Methods	*IT*	x_n	$f(x_n)$	δ	*COC*
NM	5	1.4044916482153412260350868	-4.87943e-34	1.58381e-17	2
NN1	3	1.4044916482153412260350868	0	4.15448e-39	6.04
NK	3	1.4044916482153412260350868	0	4.04252e-30	5.07
Alg. 2.4	3	1.4044916482153412260350868	0	3.13717e-40	6.04
Alg. 2.5	3	1.4044916482153412260350868	0	1.95800e-56	7.04
Alg. 2.6	3	1.4044916482153412260350868	0	1.80470e-48	7.06
Alg. 2.7	D	1.4044916482153412260350868	0	1.58381e-17	8.01
Alg. 2.8	D	1.4044916482153412260350868	0	4.93796e-25	8.0
Alg. 2.9	3	1.4044916482153412260350868	0	1.84366e-30	9
Alg. 2.10	3	1.4044916482153412260350868	0	2.01280e-55	9.13
JM1	3	1.4044916482153412260350868	0	1.58381e-17	4.03
JM2	3	1.4044916482153412260350868	0	4.93796e-25	5.05

LJM	3	1.404491648215341226035086 8	0	4.76144e-19	5.12
SM	3	1.4044916482153412260350868	0	5.69635e-21	5.09
CM1	3	1.4044916482153412260350868	0	3.75135e-25	5.04
CM2	3	1.4044916482153412260350868	0	9.56327e-36	6.03
CM3	3	1.4044916482153412260350868	0	4.82529e-35	4.86

Here 'D' for divergent.

Example 3 Consider the equation $f_3(x) = x^2 - e^x - 3x + 2, \quad x_0 = 2.$

Table 3 (Approximate solution of example 3)

Methods	*IT*	x_n	$f(x_n)$	δ	*COC*
NM	6	0.2575302854398607604553673	2.92590e-55	9.10261e-28	2
NN1	4	0.2575302854398607604553673	0	2.00000e-60	3.48
NK	4	0.2575302854398607604553673	0	4.15360e-40	5.02
Alg. 2.4	3	0.2575302854398607604553673	1.00000e-59	9.87326e-22	5.83
Alg. 2.5	3	0.2575302854398607604553673	-1.00000e-59	4.27962e-28	7.07
Alg. 2.6	3	0.2575302854398607604553673	0	1.93888e-26	6.84
Alg. 2.7	3	0.2575302854398607604553673	0	1.57867e-40	7.86
Alg. 2.8	3	0.2575302854398607604553673	-1.00000e-59	5.48794e-28	8.31
Alg. 2.9	3	0.2575302854398607604553673	-1.00000e-59	1.00000e-59	-
Alg. 2.10	3	0.2575302854398607604553673	0	3.49682e-42	4.98
JM1	4	0.2575302854398607604553673	0	1.90768e-29	5.05
JM2	4	0.2575302854398607604553673	1.00000e-59	1.00000e-59	-
LJM	4	0.2575302854398607604553673	1.00000e-59	1.00000e-59	-

SM	4	0.257530285439860760455367	0	5.69635e-21	5.09
CM1	4	0.257530285439860760455367	0	3.75135e-25	5.04
CM2	4	0.257530285439860760455367	0	9.56327e-36	6.03
CM3	4	0.257530285439860760455367	0	4.82529e-35	4.86

Example 4 . Consider the equation $f_4(x) = \cos x - x, \quad x_0 = 1.7.$

Table 4 (Approximate solution of example 4)

Methods	*IT*	x_n	$f(x_n)$	δ	*COC*
NM	5	0.739085133215160641655312	-2.03197e-32	2.34491e-16	1.99
NN1	3	0.739085133215160641655312	-1.00000e-60	5.72241e-34	5.60
NK	3	0.739085133215160641655312	-1.00000e-60	2.50589e-21	4.66
Alg. 2.4	3	0.739085133215160641655312	-1.00000e-60	2.24754e-35	5.68
Alg. 2.5	3	0.739085133215160641655312	1.00000e-60	7.47870e-52	6.68
Alg. 2.6	3	0.739085133215160641655312	1.00000e-60	1.43026e-44	6.63
Alg. 2.7	3	0.739085133215160641655312	1.00000e-60	3.00000e-60	7.62
Alg. 2.8	3	0.739085133215160641655312	1.00000e-60	1.00000e-60	-
Alg. 2.9	3	0.739085133215160641655312	-1.00000e-60	1.00000e-60	9
Alg. 2.10	3	0.739085133215160641655312	-1.00000e-60	1.00000e-60	-
JM1	3	0.739085133215160641655312	-1.00000e-60	2.34491e-16	3.60
JM2	3	0.739085133215160641655312	-1.00000e-60	5.83737e-24	4.58
LJM	3	0.739085133215160641655312	1.00000e-60	1.70292e-22	4.45
SM	3	0.739085133215160641655312	1.00000e-60	2.24183e-21	4.48

CM1	3	0.739085133215160641655312 1	-1.00000e-60	3.28792e-23	4.75
CM2	3	0.739085133215160641655312 1	1.00000e-60	2.52820e-24	5.90
CM3	3	0.739085133215160641655312 1	-1.00000e-60	2.17244e-17	4.79

Example 5 . Consider the equation $f_5(x) = (x-1)^3 - 1, \quad x_0 = 2.5.$

Table 5 (Approximate solution of example 5)

Methods	*IT*	x_n	$f(x_n)$	δ	*COC*
NM	7	2	5.03100e-56	1.29484e-28	2
NN1	3	2	0	1.80249e-17	5.79
NK	3	2	0	4.90379e-24	5
Alg. 2.4	3	2	0	1.80249e-17	5.69
Alg. 2.5	3	2	0	1.99380e-29	6.62
Alg. 2.6	3	2	0	1.18823e-20	6.54
Alg. 2.7	3	2	0	1.20336e-27	7.65
Alg. 2.8	3	2	0	5.73338e-25	7.72
Alg. 2.9	3	2	0	3.97550e-36	8.68
Alg. 2.10	3	2	0	7.32429e-34	8.61
JM1	4	2	0	1.29484e-28	4
JM2	4	2	0	2.35729e-49	5
LJM	4	2	0	9.03074e-34	4.98
SM	4	2	0	6.68430e-38	5
CM1	4	2	0	4.31570e-35	5

CM2	4	2	0	0	-
CM3	4	2	0	3.52600e-55	5

Example 6. Consider the equation $f_6(x) = x^3 - 10, \quad x_0 = 2.$

Table 6 (Approximate solution of example 6)

Methods	*IT*	x_n	$f(x_n)$	δ	*COC*
NM	5	2.15443469003188372175929 36	3.29189e-35	2.25681e-18	2
NN1	3	2.15443469003188372175929 36	-8.00000e-59	4.50228e-42	6.02
NK	3	2.15443469003188372175929 36	1.00000e-58	4.89248e-30	5.04
Alg. 2.4	3	2.15443469003188372175929 36	-8.00000e-59	4.50228e-42	6.02
Alg. 2.5	3	2.15443469003188372175929 36	1.00000e-58	1.40000e-58	7.02
Alg. 2.6	3	2.15443469003188372175929 36	1.00000e-58	2.61905e-51	7.04
Alg. 2.7	3	2.15443469003188372175929 36	-8.00000e-59	8.00000e-59	-
Alg. 2.8	3	2.15443469003188372175929 36	-8.00000e-59	8.00000e-59	-
Alg. 2.9	3	2.15443469003188372175929 36	1.00000e-58	1.00000e-58	-
Alg. 2.10	3	2.15443469003188372175929 36	-8.00000e-59	8.00000e-59	-
JM1	3	2.15443469003188372175929 36	0	1.29484e-28	4
JM2	3	2.15443469003188372175929 36	1.00000e-58	1.64194e-26	5.04
LJM	3	2.15443469003188372175929 36	-8.00000e-59	1.51742e-20	5.09
SM	3	2.15443469003188372175929 36	1.00000e-58	1.44631e-22	5.06
CM1	3	2.15443469003188372175929 36	-8.00000e-59	1.89488e-26	5.02

CM2	3	2.1544346900318837217592936	1.00000e-58	2.54320e-38	6.02
CM3	3	2.1544346900318837217592936	-8.00000e-59	2.39906e-30	4.99

Example 7 . Consider the equation $f_7(x) = e^{x^2+7x-30} - 1, \quad x_0 = 3.2.$

Table 7 (Approximate solution of example 7)

Methods	*IT*	x_n	$f(x_n)$	δ	*COC*
NM	9	3	1.37562e-53	4.01112e-28	2
NN1	4	3	0	5.44868e-34	5.98
NK	5	3	0	4.21818e-37	5.00
Alg. 2.4	4	3	0	5.32929e-28	5.95
Alg. 2.5	4	3	0	1.00000e-59	5.29
Alg. 2.6	4	3	0	1.49543e-34	6.90
Alg. 2.7	4	3	0	9.71210e-52	7.97
Alg. 2.8	4	3	0	9.31132e-40	7.98
Alg. 2.9	3	3	0	2.19280e-55	9
Alg. 2.10	3	3	0	3.36801e-24	8.52
JM1	5	3	0	4.01112e-28	4
JM2	5	3	0	1.52828e-54	5
LJM	3	3	2.00000e-58	4.86190e-25	4.96
SM	5	3	0	3.73390e-31	4.99

CM1	7	3	0	1.99393e-30	5
CM2	5	3	0	0	-
CM3	5	3	0	2.19280e-55	5

5 Conclusions.

In this paper, we have suggested new higher-order iterative methods free from second derivative for solving nonlinear equation $f(x)=0$. We have discussed the efficiency index and computational order of convergence of these new methods. Several examples are given to illustrate the efficiency of Algorithm 2.4 to Algorithm 2.10. Using the idea of this paper, one can suggest and analyze higher-order multi-step iterative methods for solving nonlinear equations. Results proved in this paper may stimulate further research.

References

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

[2] R. L. Burden and J.D. Faires, Numerical Analysis, PWS Publishing Company, Bostan, 2001.

[3] C. Chun, Iterative methods improving Newton's method by the decomposition method, *Comput. Math. Appl*. **50** (2005), pp. 1559–1568

[4] C. Chun, Some improvements of Jarrat's methods with sixth order convergences. *Appl.Math. Comput*., **190** (2007), 1432–1437.

[5] A. S. Housholder, The Numerical Treatment of a Single Nonlinear Equation, McGraw-Hill, New York, 1970.

[6] J.F. Traub, Iterative Methods for Solution of Equations, Prentice-Hall, Englewood Cliffs, NJ, 1964.

[7] Y. M. Ham, C. Chun and S. G.Lee, Some higher-order modifications of Newton's method for solving nonlinear equations, *J. Comput. Appl. Math.* **222** (2008) 477-486.

[8] M. Javidi, Fourth-order and fifth-order iterative methods for nonlinear algebraic equations, *Math. Comput. Model*., **50** (2009) 66-71.

[9] Y. T. Li and A. Q. Jiao, Some variants of Newton's method with fifth-order and fourth-order convergence for solving nonlinear equations, *Int. J. Appl. Math. Comput*., **1** (2009) 1-16.

[10] M. A. Noor, New family of iterative methods for nonlinear equations, Appl. Math. Comput. **190** (2007), pp. 553–558.

[11] M. A. Noor and K. I. Noor, Iterative schemes for solving nonlinear equations, *Appl. Math. Comput*. **183** (2006), pp. 774–779.

[12] M. A. Noor and K. I. Noor, Three-step iterative methods for nonlinear equations, *Appl. Math. Comput*. **183** (2006), pp. 322–327.

[13] K. I. Noor, M. A. Noor and S. Momani, Modified householder iterative method for nonlinear equations, *Appl. Math. Comput*. **190** (2007), pp. 1534–1539.

[14] M. A. Noor and K. I. Noor, Predicot-corrector Halley method for nonlinear equations,*Appl. Math. Comput*. **188** (2007), pp. 1587–1591.

[15] M. A. Noor, Some iterative methods for solving nonlinear equations using homotopy perturbation method, Int. J. Comp. Math., **87** (2010) 141-149.

[16] M. A. Noor, W. A. Khan, A. Hussain, A new modified Halley method without second derivatives for nonlinear equation, *Appl. Math. Comput*., **189** (2007) 1268-1273.

[17] M. A. Noor, Iterative methods for nonlinear equations using homotopy perturbation technique, *Appl. Math. Inform. Sci.* **4** (2) (2010) 227–235

[18] W. Gautschi, Numerical Analysis: An introduction, Birkhauser, 1997.

[19] H. I. Siyyam, An iterative method with fifth-order convergence for nonlinear equations, *Appl. Math. Sci.,* **3** (2009) 2041-2053.

[20] M. A. Noor, W. A. Khan, K. I. Noor and E. S. Said, (2011). Higher-order iterative methods free from second derivative for solving nonlinear equations. *Int. J. Phy. Sci.* **6(8**), 1887-1897

[21] M. A. Noor, W. A. Khan, S. Younus Homotopy perturbation technique for solving certain nonlinear equations, *Appl. Math. Sci*., Vol. **6,** 2012, no. 130, 6487 – 6499.

[22] M. A. Noor, W. A. Khan, Fourth-Order Iterative Method Free from Second Derivative for Solving Nonlinear Equations, *Appl. Math. Sci*., Vol**. 6**, 2012, no. 93, 4617 – 4625.

[23] M. A. Noor, W. A. Khan, New iterative methods for solving nonlinear equation by using homotopy perturbation method, *Appl. Math. Comput*., **219** (2012) pp. 3565-3574.