ISSN:0975-9646



# Iterative Methods of Order Four for Solving Nonlinear Equations

V.B. Kumar,Vatti[1], Shouri Dominic[2] and Mounica,V[3]

*Department of Engineering Mathematics[1,2]*
*Former Student of Chemical Engineering[3]*
*Andhra University College of Engineering (A), Andhra University*
*Visakhapatnam – 530003, Andhra Pradesh, India*

*Abstract* – **In this paper, we suggest an iterative method of order four for solving nonlinear equations and some more are derived from this new method. The efficiency index of this method is $\sqrt[3]{4}$. Several examples are considered and compared with the existing methods.**

***Keywords*** **- Iterative Method, Nonlinear Equations, Convergence Criteria, Numerical Examples, Newton's Method.**

## I. Introduction

The well known quadratic convergent Newton's method for finding a simple root of the non-linear equation

$$f(x) = 0 \qquad (1.1)$$

Where $f : D \subset R \to R$ is a scalar function on an open interval D, is given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \qquad (1.2)$$

(n = 0, 1, 2, 3 ………)

Many iterative methods have been developed see [1-13] for solving the equation (1.1) by using several techniques including perturbation methods and quadrature formulae. Noor [8] suggested the following algorithm which has fourth order convergence

For a given $x_0$, Noor's two step algorithm to compute $x_{n+1}$ is

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)} \qquad (1.3)$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}\left[\frac{f(x_n) - f(y_n)}{f(x_n) - 2f(y_n)}\right] \qquad (1.4)$$

Recently, Jafar and Behzad [6] derived few variants of King's fourth order family [12]

i.e,

$$x_{n+1} = y_n - \left[\frac{f(x_n) + \beta f(y_n)}{f(x_n) + (\beta - 2) f(y_n)}\right]\frac{f(y_n)}{f'(x_n)} \qquad (1.5)$$

Where $y_n$ is as given in (1.3)

And, some of the variants suggested by Jafar and Behzad [6] are

i. $$x_{n+1} = x_n - \left[1 + \frac{f(y_n)}{f(x_n)} + 2\left(\frac{f(y_n)}{f(x_n)}\right)^2\right]\frac{f(x_n)}{f'(x_n)} \qquad (1.6)$$

ii. $$x_{n+1} = x_n - \left[1 + \frac{f(y_n)}{f(x_n)} + 2\left(\frac{f(y_n)}{f(x_n)}\right)^2 + \left(\frac{f(y_n)}{f(x_n)}\right)^3\right] \times \frac{f(x_n)}{f'(x_n)} \qquad (1.7)$$

iii. $$x_{n+1} = x_n - \left[\frac{f(x_n) + f(y_n)}{f^2(x_n) - 2f^2(y_n)}\right]\frac{f^2(x_n)}{f'(x_n)} \qquad (1.8)$$

iv. $$x_{n+1} = y_n - \left[\frac{f(x_n) + f(y_n)}{f(x_n) - f(y_n)}\right]\frac{f(y_n)}{f'(x_n)} \qquad (1.9)$$

All the above formulae are having fourth order convergence and $y_n$ is as given in (1.3) only.

In the section 2 of this paper, we develop an iterative method for solving (1.1) and its convergence criteria is discussed. And also, few variants are derived from this new method in the same section. Several numerical examples are considered and compared with existing ones in the concluding section.

## II. The New Iterative Method

Let $'\alpha'$ be the exact root of the equation (1.1) in an open interval D in which $f(x)$ is continuous and has well defined first derivative and let $x_n$ be the $n^{th}$ approximate to the root

of (1.1) and

$$\alpha = x_n + e_n \qquad (2.1)$$

Where $e_n$ is the error at the $n^{th}$ stage and

$$f(\alpha) = 0 \qquad (2.2)$$

(1.5)

Expanding $f(\alpha)$ by Taylor's series about $x_n$, we have

$$f(\alpha) = f(x_n) + (\alpha - x_n) f'(x_n) + \frac{(\alpha - x_n)^2}{2} f''(x_n) + \ldots\ldots\ldots\ldots \qquad (2.3)$$

Assume $e_n$ is small enough and neglecting the higher powers in (2.3), from (2.1) & (2.2)

We have

$$e_n^2 f''(x_n) + 2e_n f'(x_n) + 2f(x_n) = 0 \qquad (2.4)$$

Which yields us

$$e_n = -2\frac{f(x_n)}{f'(y_n)}\left[\frac{1}{1\pm\sqrt{1-\frac{4f(y_n)}{f(x_n)}}}\right] \qquad (2.5)$$

To make the denominator largest in magnitude, we take $e_n$ as

$$e_n = -2\frac{f(x_n)}{f'(y_n)}\left[\frac{1}{1+\sqrt{1-\frac{4f(y_n)}{f(x_n)}}}\right] \qquad (2.6)$$

Taking $'\alpha'$ in (2.1) as $(n+1)^{th}$ approximate to the root, from (2.1) and (2.6), we now define the following algorithm.

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

$$x_{n+1} = x_n - 2\frac{f(x_n)}{f'(y_n)}\left[\frac{1}{1+\left(1-\frac{4f(y_n)}{f(x_n)}\right)^{\frac{1}{2}}}\right] \qquad (2.7)$$

(n = 0, 1, 2, 3 ……….)

Where $y_n$ is as given in (1.3)

This algorithm is free from second derivative and requires two functional evaluations and one of its first derivatives. The efficiency index of this method is $\sqrt[3]{4}$.

**Theorem 2.1**: Let $\alpha \in D$ be a single zero of sufficiently differentiable function $f: D \subset R \to R$ for an open interval D. If $x_0$ is in the vicinity of $\alpha$, then algorithm 2.1 has fourth order convergence.

**Proof:** If $'\alpha'$ be the root and $x_n$ be the $n^{th}$ approximate to the root, then expanding $f(x_n)$ about $'\alpha'$ using Taylor's expansion, we have

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

$$= f'(\alpha)\left[e_n + \frac{1}{2!}\frac{f''(\alpha)}{f'(\alpha)}e_n^2 + \frac{1}{3!}\frac{f'''(\alpha)}{f'(\alpha)}e_n^3 + \frac{1}{4!}\frac{f^{iv}(\alpha)}{f'(\alpha)}e_n^4 + \frac{1}{5!}\frac{f^{v}(\alpha)}{f'(\alpha)}e_n^5 + O(e_n^6)\right]$$

$$= f'(\alpha)\left[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + c_5e_n^5 + O(e_n^6)\right] \qquad (2.8)$$

Where $c_j = \frac{1}{j!}\frac{f^j(\alpha)}{f'(\alpha)}$, ($j$=2, 3, 4…)

And,

$$f'(x_n) = f'(\alpha)\left[1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + 5c_5e_n^4 + O(e_n^5)\right] \qquad (2.9)$$

Now,

$$\frac{f(x_n)}{f'(x_n)} = \left[e_n - c_2e_n^2 - (2c_3 - 2c_2^2)e_n^3 - (3c_4 - 7c_2c_3 + 4c_2^3)e_n^4 + O(e_n^5)\right] \qquad (2.10)$$

From (1.3) and (2.10), we have

$$y_n = \left[\alpha + c_2e_n^2 + (2c_3 - 2c_2^2)e_n^3 + (3c_4 - 7c_2c_3 + 4c_2^3)e_n^4 + O(e_n^5)\right] \qquad (2.11)$$

$$f(y_n) = f'(\alpha)\left[c_2e_n^2 - (2c_2^2 - 2c_3)e_n^3 - (7c_2c_3 - 5c_2^3 - 3c_4)e_n^4 + O(e_n^5)\right] \qquad (2.12)$$

$$\frac{f(y_n)}{f(x_n)} = \frac{f'(\alpha)\left[c_2e_n^2 - (2c_2^2 - 2c_3)e_n^3 - (7c_2c_3 - 5c_2^3 - 3c_4)e_n^4 + O(e_n^5)\right]}{f'(\alpha)\left[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + c_5e_n^5 + O(e_n^6)\right]}$$

$$= [c_2e_n^2 - (2c_2^2 - 2c_3)e_n^3 - (7c_2c_3 - 5c_2^3 - 3c_4)e_n^4 + O(e_n^5)] \times [1 - c_2e_n + (c_2^2 - c_3)e_n^2 + (2c_2c_3 - c_2^3 - c_4)e_n^3 + O(e_n^4)]$$

$$= \left[\begin{array}{l} c_2e_n + (2c_3 - 2c_2^2)e_n^2 + (5c_2^3 - 7c_2c_3 + 3c_4)e_n^3 - c_2^2e_n^2 \\ +(2c_2^3 - 2c_2c_3)e_n^3 - (5c_2^4 + 3c_2c_4 - 7c_2^2c_3)e_n^4 \\ +(c_2^3 - c_2c_3)e_n^3 + (2c_3 - 2c_2^2)(c_2^2 - c_3)e_n^4 \\ +(2c_2^2c_3 - c_2c_4 - c_2^4)e_n^4 + O(e_n^5) \end{array}\right]$$

Thus,

$$\frac{f(y_n)}{f(x_n)}=\begin{bmatrix} c_2e_n+(2c_3-3c_2^2)e_n^2+(8c_2^3-10c_2c_3+3c_4)e_n^3 \\ +O(e_n^4)\end{bmatrix} \quad (2.13)$$

From (2.13), we obtain

$$1-4\frac{f(y_n)}{f(x_n)}=1-4\begin{bmatrix} c_2e_n+(2c_3-3c_2^2)e_n^2+(8c_2^3-10c_2c_3 \\ +3c_4)e_n^3+O(e_n^4)\end{bmatrix} \quad (2.14)$$

$$\left(1-4\frac{f(y_n)}{f(x_n)}\right)^{\frac{1}{2}}=1+\frac{1}{2}\left\{-4\begin{bmatrix} c_2e_n+(2c_3-3c_2^2)e_n^2+ \\ (8c_2^3-10c_2c_3+3c_4)e_n^3\end{bmatrix}\right\}$$
$$-\frac{1}{8}\left\{16\begin{bmatrix} c_2^2e_n^2+(2c_3-3c_2^2)^2e_n^4+ \\ 2c_2(2c_3-3c_2^2)e_n^3 \\ +2c_2(8c_2^3-10c_2c_3+3c_4)e_n^4\end{bmatrix}\right\}$$
$$-\frac{3}{48}\left\{64\begin{bmatrix} c_2^3e_n^3+2c_2^2(2c_3-3c_2^2) \\ +(2c_3-3c_2^2)c_2^2e_n^4\end{bmatrix}\right\}$$
$$-\frac{15}{384}\{256[c_2^4e_n^4]\}$$

$$=1-2\left[c_2e_n+(2c_3-3c_2^2)e_n^2+(8c_2^3-10c_2c_3+3c_4)e_n^3\right]$$
$$-2\begin{bmatrix} c_2^2e_n^2+(4c_2c_3-6c_2^3)e_n^3+ \\ (4c_3^2-12c_2^2c_3+4c_2^4+16c_2^4-20c_2^2c_3+6c_2c_4)e_n^4\end{bmatrix}$$
$$-4\left[c_2^3e_n^3+(4c_2^2c_3-9c_2^4)e_n^4\right]$$
$$-10[c_2^4e_n^4]$$
$$=\begin{bmatrix} 1-2c_2e_n+(4c_2^2-4c_3)e_n^2+(12c_2c_3-8c_2^3-6c_4)e_n^3 \\ +O(e_n^4)\end{bmatrix} \quad (2.15)$$

$$1+\left(1-4\frac{f(y_n)}{f(x_n)}\right)^{\frac{1}{2}}=2\begin{bmatrix} 1-c_2e_n+(2c_2^2-2c_3)e_n^2+ \\ (6c_2c_3-4c_2^3-3c_4)e_n^3+O(e_n^4)\end{bmatrix} \quad (2.16)$$

Thus,

$$2\frac{f(x_n)}{f'(x_n)}\left[1+\left(1-4\frac{f(y_n)}{f(x_n)}\right)^{\frac{1}{2}}\right]^{-1}$$

$$=\frac{2\begin{bmatrix} e_n-c_2e_n^2-(2c_3-2c_2^2)e_n^3-(3c_4-7c_2c_3+4c_2^3)e_n^4 \\ +O(e_n^5)\end{bmatrix}}{2\begin{bmatrix} 1-c_2e_n+(2c_2^2-2c_3)e_n^2+(6c_2c_3-4c_2^3-3c_4)e_n^3 \\ +O(e_n^4)\end{bmatrix}}$$

$$=\begin{bmatrix} e_n-c_2e_n^2-(2c_3-2c_2^2)e_n^3-(3c_4-7c_2c_3+4c_2^3)e_n^4 \\ +O(e_n^5)\end{bmatrix}$$
$$\times\begin{bmatrix} 1-\{c_2e_n+(2c_3-2c_2^2)e_n^2+(4c_2^3+3c_4-6c_2c_3)e_n^3 \\ +O(e_n^4)\end{bmatrix}^{-1}$$

$$=\begin{bmatrix} e_n-c_2e_n^2-(2c_3-2c_2^2)e_n^3-(3c_4-7c_2c_3+4c_2^3)e_n^4 \\ +O(e_n^5)\end{bmatrix}$$
$$\times\begin{bmatrix} 1+c_2e_n+(2c_3-2c_2^2)e_n^2+(4c_2^3+3c_4-6c_2c_3)e_n^3 \\ +(c_2^2e_n^2+(2c_3-2c_2^2)^2e_n^4+2c_2(2c_3-2c_2^2)e_n^3 \\ +2c_2(4c_2^3+3c_4-6c_2c_3)e_n^4+(c_2^3e_n^3 \\ +2c_2^2(2c_3-2c_2^2)e_n^4+c_2^2(2c_3-2c_2^2)e_n^4+c_2^4e_n^4\end{bmatrix}$$

$$=\begin{bmatrix} e_n-c_2e_n^2-(2c_3-2c_2^2)e_n^3-(3c_4-7c_2c_3+4c_2^3)e_n^4 \\ +O(e_n^5)\end{bmatrix}$$
$$\begin{bmatrix} 1+c_2e_n+(2c_3-c_2^2)e_n^2+(4c_2^3+3c_4-6c_2c_3 \\ +4c_2c_3-4c_2^3+c_2^3)e_n^3+O(e_n^4)\end{bmatrix}$$

$$=\begin{bmatrix} e_n+c_2e_n^2+2c_3e_n^3-c_2^2e_n^3+(c_2^3+3c_4-6c_2c_3)e_n^4 \\ -c_2e_n^2-c_2^2e_n^3-(2c_2c_3+c_2^3)e_n^4-(2c_3-2c_2^2)e_n^3 \\ -c_2(2c_3-2c_2^2)e_n^4-(3c_4-7c_2c_3+4c_2^3)e_n^4+O(e_n^5)\end{bmatrix}$$

$$=\begin{bmatrix} e_n+(c_2-c_2)e_n^2+(2c_3-c_2^2-c_2^2+2c_2^2-2c_3)e_n^3 \\ +(2c_2^3-c_2^3-4c_2^3+c_2^3+3c_4-3c_4-2c_2c_3-2c_2c_3 \\ -2c_2c_3+7c_2c_3)e_n^4+O(e_n^5)\end{bmatrix}$$

$$=e_n+(c_2c_3-2c_2^3)e_n^4+O(e_n^5) \quad (2.17)$$

From (2.1), (2.7) and (2.17) we have the rate of convergence of the method (2.7) is four.

**Case 2.1:** By expanding $\sqrt{1-4\frac{f(y_n)}{f(x_n)}}$ appearing in the denominator of the method (2.7), we obtain

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}\left\{\frac{1}{1-\left(\frac{f(y_n)}{f(x_n)}+\left(\frac{f(y_n)}{f(x_n)}\right)^2+2\left(\frac{f(y_n)}{f(x_n)}\right)^3\right)}\right\} \quad (2.18)$$

And, the above further yields

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}\left[1+\frac{f(y_n)}{f(x_n)}+2\left(\frac{f(y_n)}{f(x_n)}\right)^2+5\left(\frac{f(y_n)}{f(x_n)}\right)^3\right] \quad (2.19)$$

Considering up to first degree, second degree and third degree terms of the expression lying within the brackets of the formula (2.19), we have the following algorithms.

**Algorithm 2.2:** For a given $x_0$, compute the approximate solution $x_{n+1}$ by iterative scheme.

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}\left[1+\frac{f(y_n)}{f(x_n)}\right] \quad (2.20)$$

(n = 0, 1, 2, 3 ……..)

Where $y_n$ is as given in (1.3).

**Theorem 2.2:** Let $\alpha\in D$ be a simple zero of sufficiently differentiable function $f:D\subset R\to R$ for an open interval D. If $x_0$ is in the vicinity of $\alpha$ , then Algorithm 2.2 has third order convergence.

**Proof:** As done in theorem (2.1), one can easily obtain the error relation as

$$\alpha+e_{n+1}=\alpha+e_n-\left[\begin{array}{l}e_n-2c_2^2e_n^3+(4c_2^3-14c_2c_3+3c_4)e_n^4\\+O(e_n^5)\end{array}\right]$$

Which gives us

$$e_{n+1}\propto e_n^3 \quad (2.21)$$

Therefore, the algorithm (2.2) has third order convergence.

**Algorithm 2.3:** For a given $x_0$, compute the approximate solution $x_{n+1}$ by iterative scheme.

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}\left[1+\frac{f(y_n)}{f(x_n)}+2\left(\frac{f(y_n)}{f(x_n)}\right)^2\right] \quad (2.22)$$

(n = 0, 1, 2, 3 ……..)

Where $y_n$ is as given in (1.3).

**Theorem 2.3:** Let $\alpha\in D$ be a simple zero of sufficiently differentiable function $f:D\subset R\to R$ for an open interval D. If $x_0$ is in the vicinity of $\alpha$ , then Algorithm 2.3 has fourth order convergence.

**Proof:** As done in theorem (2.1), one can easily obtain the error relation as

$$\alpha+e_{n+1}=\alpha+e_n-[e_n+(c_2c_3-5c_2^3)e_n^4+O(e_n^5)]$$

Which gives us

$$e_{n+1}\propto e_n^4 \quad (2.23)$$

Therefore, the algorithm (2.3) has fourth order convergence.

**Algorithm 2.4:** For a given $x_0$, compute the approximate solution $x_{n+1}$ by iterative scheme.

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}\left[1+\frac{f(y_n)}{f(x_n)}+2\left(\frac{f(y_n)}{f(x_n)}\right)^2+5\left(\frac{f(y_n)}{f(x_n)}\right)^3\right] \quad (2.24)$$

(n = 0, 1, 2, 3 ……..)

Where $y_n$ is as given in (1.3).

**Theorem 2.4:** Let $\alpha\in D$ be a simple zero of sufficiently differentiable function $f:D\subset R\to R$ for an open interval D. If $x_0$ is in the vicinity of $\alpha$ , then Algorithm 2.4 has fourth order convergence.

**Proof:** As done in theorem (2.1), one can easily obtain the error relation as

$$\alpha+e_{n+1}=\alpha+e_n-[e_n+(c_2c_3+4c_2^3)e_n^4+O(e_n^5)]$$

Which gives us

$$e_{n+1}\propto e_n^4 \quad (2.25)$$

Therefore, the algorithm (2.4) has fourth order convergence.

## III. Numerical Examples

In this section, several examples are considered which are taken from [6, 11] and all the methods presented in this paper are being taken to tabulate the computational results below by using the stopping criteria $|f(x_n)|<10^{-15}$.

TABLE. I

| Equation and its root | Initial Guess $x_0$ | Number of iterations taken to obtain the root by applying the following methods | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | 1.2 | 1.4 | 1.5 | 1.6 | 1.7 | 1.8 | 1.9 | 2.7 | 2.19 |
| 1. $\sin^2 x - x^2 + 1 = 0$, $x = 1.40449165$ | -4 | 6 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 |
| | -1 | 6 | 3 | 7 | 4 | 4 | 8 | 4 | 3 | 4 |
| | 1 | 6 | 3 | 7 | 4 | 4 | 8 | 4 | 3 | 4 |
| | 1.3 | 4 | 2 | 3 | 3 | 3 | 3 | 3 | 2 | 2 |
| | 2 | 5 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| | 3 | 6 | 3 | 4 | 4 | 4 | 4 | 4 | 3 | 3 |
| | 5 | 7 | 4 | 4 | 4 | 4 | 4 | 4 | Err | 3 |
| 2. $x^2 - e^x - 3x + 2 = 0$, $x = 0.25753029$ | -5 | 6 | 3 | 4 | 4 | 4 | 3 | 3 | 3 | 3 |
| | -1 | 5 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 |
| | 0 | 4 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 1 | 4 | 3 | 2 | 2 | 2 | 2 | 2 | 3 | 3 |
| | 2 | 5 | 3 | 3 | 3 | 3 | 3 | 3 | Err | 3 |
| 3. $\cos x - x = 0$, $x = 0.73908513$ | 0 | 6 | 3 | 4 | 3 | 3 | 3 | 3 | 3 | 3 |
| | 1 | 4 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 1.7 | 4 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 4. $xe^{x^2} - \sin^2 x + 3\cos x + 5 = 0$, $x = -1.20764783$ | -1 | 6 | 4 | 5 | 5 | 4 | 4 | 4 | 4 | 5 |
| | -2 | 9 | 5 | 6 | 6 | 6 | 5 | 6 | Err | 5 |
| 5. $x^2 \sin^2 x + e^{x^2 \sin x \cos x} - 28 = 0$, $x = 3.43717174$, $4.62210416$ | 4 | 7 | 5 | 6 | 5 | 5 | 5 | 5 | 5 | 5 |
| | 4.5 | 8 | 5 | 5 | 5 | 5 | 5 | 5 | Err | 5 |
| 6. $x^3 + 4x^2 - 10 = 0$, $x = 1.36523001$ | -0.5 | 142 | 14 | 11 | 15 | 19 | 56 | 16 | Err | 49 |
| | -0.3 | 54 | 71 | 48 | 27 | 59 | 9 | 7 | Err | 17 |
| | 1 | 5 | 3 | 4 | 4 | 3 | 3 | 3 | 3 | 4 |
| | 1.5 | 4 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 |
| | 2 | 5 | 3 | 3 | 4 | 3 | 3 | 3 | 3 | 3 |
| | 3 | 6 | 3 | 4 | 4 | 4 | 4 | 4 | 3 | 3 |

## IV. CONCLUSION

The tabulated results show that the methods (1.4) to (2.19) are converging at almost same pace compared to the method (1.2). And, the algorithm (2.7) is also working well except in the case that $\left(1 - 4\frac{f(y_n)}{f(x_n)}\right)$ is negative.

## REFERENCES


[1] A. M. Ostrowski, Solution of Equations and Systems of Equations, Academic Press, New York, 1966.

[2] C. Chun, Iterative Methods Improving Newton's Method by The Decomposition Method, Comput. Math. Appl. 50(2005) 1559-1568.

[3] J. F. Traub, Iterative Methods for The Solution Of Equations, Prentice Hall, Englewood Cliffs (NJ), 1964.

[4] J. Kou, Second – Derivative – Free Variants of Cauchy's Method, Appl. Math. Comput. (2007) in press.

[5] J.S. Kou, Y.T. Li and X.H. Wang, Modified Halley's Method Free from Second Derivative, Appl. Math. Comput. (2006), in press.

[6] Jafar Biazar, Behzad Ghanbari, A General Fourth – order Family of Methods for Solving Nonlinear Equations, MACMESE'09 Proceedings of the 11th WSEAS international conference on Mathematical and computational methods in science and engineering (2009), 79-82.

[7] M. A. Noor, W. A. Khan and Akhtar Hussain, A New Modified Halley Method Without Second Derivatives for Nonlinear Equation, Appl. Math. Comput., 189(2007) 1268-1273.

[8] M. A. Noor, W. A. Khan, Fourth – Order Iterative Method Free from Second Derivative for Solving Nonlinear Equations, Applied Mathematical Sciences, Vol. 6, 2012, no. 93, 4617-4625.

[9] M. A. Noor, W. A. Khan, K. I. Noor and E. A. Said, Higher – order Iterative Methods Free from Second Derivative for Solving Nonlinear Equations, Inter. J. Phys. Sci. 6(8) (2011) 1887-1893.

[10] M. Aslam Noor and V. Gupta, Modified Householder Iterative Method Free from Second Derivatives for Nonlinear Equations, Appl. Math. Comput., (2007), In press.

[11] M. Aslam Noor, Numerical Analysis and Optimization, Lecture Notes, Mathematics Department, COMSATS Institute of Information Technology, Islamabad, Pakistan, 2006-2011.

[12] R. King, Family of Fourth-Order Methods for Nonlinear Equations, SIAM J. Numer. Anal. 10(5)(1973), 876-879.

[13] S. Abbasbandy, Improving Newton-Raphson Method for Nonlinear Equations by Modified Adomian Decomposition Method, Appl. Math. Comput. 145 (2003) 887-893.