Kragujevac Journal of Mathematics
Volume 50(4) (2026), Pages 513–528.

UNI- AND BI-PARAMETRIC TWO-STEP ITERATIVE METHOD WITH MEMORY FOR SOLVING NONLINEAR EQUATIONS

NISHANT KUMAR[1] AND JAI PRAKASH JAISWAL[2]

Abstract. In this paper, we have suggested a two-step with memory method for solving nonlinear equations by transforming an extant optimal fourth-order without memory method. The acceleration of the order of convergence is attained by employing a single and two self-accelerating parameters. These parameters are estimated by a Hermite interpolating polynomial to enhance the convergence order of iterative method without memory. This order of convergence acceleration is achieved without the use of any additional functional evaluations, precisely the convergence order of the suggested two-step with memory method is reached from 4 to 5.70156. The rate of convergence is also verified by Herzberger's matrix method. Finally, various examples are taken into consideration to support the theoretical outcomes.

1. Introduction

In today's real world, solving the nonlinear equation $g(y) = 0$, is a very momentous problem. Numerous iterative methods have been presented to find the nonlinear equation's solution (see [1–4]). These iterative methods show a very important role in the area of numerical analysis because they are utilized in a wide range of pure and applied science fields. The most popular one-point without memory iterative technique among them is the Newton-Raphson method, which is described by

$$w_{n+1} = w_n - \frac{g(w_n)}{g'(w_n)},$$

Key words and phrases. Iterative method with memory, Hermite interpolating polynomial, R-order of convergence, nonlinear equation, root finding, computational efficiency.

2020 *Mathematics Subject Classification.* Primary: 65H05. Secondary: 65B99, 65G99.

DOI

Received: April 01, 2023.

Accepted: February 11, 2024.

for the solution of $g(y) = 0$, w_0 is the initial approximation and $n = 0, 1, 2, \ldots$, whose convergence order is 2. One issue with this method is the presumption $g'(w_n) \neq 0$, which restricts it's application. One-point iterative scheme established by Kumar et al. [5] is described as follows:

$$w_{n+1} = z_n - \frac{g(w_n)}{g'(w_n) - \lambda g(w_n)}.$$

Taking $\lambda = 0$ in the above equation, we achieve the Newton-Raphson method. The error expression of the aforesaid scheme is

$$e_{n+1} = (\lambda - c_2)e_n^2 + O(e_n^3),$$

where $e_n = w_n - \gamma$, $c_i = \frac{g^{(i)}(\gamma)}{i!g'(\gamma)}$, $i = 2, 3, \ldots$, and γ is a zero of nonlinear equation $g(w) = 0$. The convergence order of the aforesaid method can be increased by taking $\lambda = c_2$ in the above error expression. For the classification of iterative methods one can go through the references [6, 7].

Several researchers are currently concentrating on creating with memory iterative techniques that uses one or more self-accelerating parameters. There are some excellent contributions dedicated to derivative free with memory iterative techniques, such as [8–12]. Unfortunately, there are very few memory-based derivative iterative techniques for solving nonlinear equations are available in the literature. The development of the multipoint iterative technique with memory is the main goal of this paper because it may raise the order of convergence of the optimal without memory methods without requiring any additional computations and has a high computational efficiency. In this paper, we present a uni- and bi-parametric two-step iterative method with memory for solving nonlinear equations, followed by a convergence analysis. The Hermite interpolating polynomial is used to calculate the parameters, and the order of convergence of the optimal two-point method is increased from 4 to 5 and 5.70156, respectively. The convergence rate is also verified by an alternate approach called Herzberger's matrix method [13]. At the last, the derived theoretical results are validated by numerical testing.

2. With Memory Method and its Convergence Analysis

In the following part, we will add the parameter α to the iterative method presented by Khattri [14] to improve it's convergence rate. First, we take into account the fourth-order without memory method, which is given in the article [14]:

$$\begin{aligned} z_n &= w_n - \frac{g(w_n)}{g'(w_n)}, \\ w_{n+1} &= z_n - \frac{g(z_n)}{2\left(\frac{g(z_n)-g(w_n)}{z_n-w_n}\right) - g'(w_n)}. \end{aligned} \tag{2.1}$$

The error expressions for each sub-step of (2.1) are:

$$e_{n,z} = z_n - \gamma = c_2 e_n^2 + O(e_n^3),$$
$$e_{n+1} = (c_2^3 - c_2 c_3) e_n^4 + O(e_n^5),$$

where $e_{n,z} = z_n - \gamma$, $e_n = w_n - \gamma$ and $c_i = \frac{g^{(i)}(\gamma)}{i! g'(\gamma)}$, for $i = 2, 3, 4, \dots$, and $\gamma \in \mathbb{R}$. After adding the parameter α_n to the first sub-step of the above scheme, we can write the following with memory iterative scheme:

$$\begin{aligned} z_n &= w_n - \frac{g(w_n)}{g'(w_n) - \alpha_n g(w_n)}, \\ w_{n+1} &= z_n - \frac{g(z_n)}{2\left(\frac{g(z_n) - g(w_n)}{z_n - w_n}\right) - g'(w_n)}. \end{aligned} \tag{2.2}$$

The error expressions for each sub- step of (2.2) are:

$$e_{n,z} = z_n - \gamma = (-\alpha_n + c_2) e_n^2 + O(e_n^3), \tag{2.3}$$

$$e_{n+1} = (\alpha_n - c_2)((\alpha_n - c_2)c_2 + c_3) e_n^4 + O(e_n^5), \tag{2.4}$$

where $e_{n,z} = z_n - \gamma$, $e_n = w_n - \gamma$ and $c_i = \frac{g^{(i)}(\gamma)}{i! g'(\gamma)}$, for $i = 2, 3, 4, \dots$, and $\gamma \in \mathbb{R}$. It is symbolized by **OWM4**. It is clear from (2.4) that the order of convergence of (2.2) is four for $\alpha_n \neq c_2$ and when $\alpha_n = c_2 = \frac{g''(\gamma)}{2! g'(\gamma)}$, the convergence order of (2.2) is five. Now the issue is that the exact values of $g'(\gamma)$ and $g''(\gamma)$ are not available for this form of acceleration of convergence but we can use the data available from the most recent iteration and the one before it, and it satisfies the condition $\lim_{n\to+\infty} \alpha_n = c_2 = \frac{g''(\gamma)}{2! g'(\gamma)}$ for the asymptotic error constant to be zero in the equation (2.4). For calculating α_n, consider the best possible approximation:

$$\alpha_n = \frac{H_4''(w_n)}{2g'(w_n)}, \tag{2.5}$$

where

$$\begin{aligned} H_4(w) =& g(w_n) + (w - w_n) g[w_n, w_n] + (w - w_n)^2 g[w_n, w_n, z_{n-1}] + (w - w_n)^2 \\ & \times (w - z_{n-1}) g[w_n, w_n, z_{n-1}, w_{n-1}] + (w - w_n)^2 (w - z_{n-1})(w - w_{n-1}) \\ & \times g[w_n, w_n, z_{n-1}, w_{n-1}, w_{n-1}], \end{aligned}$$

and so,

$$\begin{aligned} H_4''(w_n) =& 2g[w_n, w_n, z_{n-1}] + (w_n - z_{n-1})(4g[w_n, w_n, z_{n-1}, w_{n-1}] \\ & - 2g[w_n, z_{n-1}, w_{n-1}, w_{n-1}]). \end{aligned}$$

Theorem 2.1. *Let a Hermite interpolating polynomial H_m of degree m which interpolates a function g at nodes $w_n, w_n, t_0, \dots, t_{m-2}$ located within an interval I, and the derivative $g^{(m+1)}$ is continuous in I, as well as the Hermite interpolating polynomial satisfying the conditions $H_m(w_n) = g(w_n)$, $H_m'(w_n) = g'(w_n)$, $H_m(t_i) = g(t_i)$,*

$i = 0, 1, \ldots, m-2$. *Indicate the errors* $e_{t,i} = t_i - \gamma$, $i = 0, 1, 2, \ldots, m-2$, *and presume that*

(1) *all nodes* $w_n, t_0, \ldots, t_{m-2}$ *are adequately near to the zero* γ*;*
(2) *the condition* $e_n = O(e_{t,0}, e_{t,1}, \ldots, e_{t,m-2})$ *holds.*

Then

$$\begin{aligned} H_m{}''(w_n) =& 2g'(\gamma)\left(c_2 - (-1)^{m-1} c_{m+1} \prod_{i=0}^{m-2} e_{t,i} + 3c_3 e_n\right), \\ \alpha_n =& \frac{H_m''(w_n)}{2g'(w_n)} \sim \left(c_2 - (-1)^{m-1} c_{m+1} \prod_{i=0}^{m-2} e_{t,i} + (3c_3 - 2c_2^2) e_n\right) \end{aligned}$$

and

$$\alpha_n - c_2 \sim \left(-(-1)^{m-1} c_{m+1} \prod_{i=0}^{m-2} e_{t,i} + (3c_3 - 2c_2^2) e_n\right).$$

Proof. The Hermite interpolation error expression can be written as follows:

$$g(w) - H_m(w) = \frac{g^{(m+1)}(\xi)}{(m+1)!}(w - w_n)^2 \prod_{i=0}^{m-2}(w - t_i), \quad \xi \in I.$$

After differentiating the aforementioned expression twice at the point $w = w_n$, we succeed

$$g''(w_n) - H_m''(w_n) = 2\frac{g^{(m+1)}(\xi)}{(m+1)!} \prod_{i=0}^{m-2}(w_n - t_i), \quad \xi \in I,$$

or

$$H_m''(w_n) = g''(w_n) - 2\frac{g^{(m+1)}(\xi)}{(m+1)!} \prod_{i=0}^{m-2}(w_n - t_i), \quad \xi \in I. \tag{2.6}$$

Using Taylor's expansion of derivative of g at the point $w_n \in I$ and $\xi \in I$ around the root γ of g gives

$$g'(w_n) = g'(\gamma)(1 + 2c_2 e_n + 3c_3 e_n^2 + O(e_n^3)), \tag{2.7}$$

$$g''(w_n) = g'(\gamma)(2! c_2 + 3! c_3 e_n + O(e_n^2)) \tag{2.8}$$

and

$$g^{(m+1)}(\xi) = g'(\gamma)((m+1)! c_{m+1} + (m+2)! c_{m+2} e_\xi + O(e_\xi^2)). \tag{2.9}$$

Putting the expressions (2.8), (2.9) in the equation (2.6), we obtain

$$H_m''(w_n) = 2g'(\gamma)\left(c_2 - (-1)^{m-1} c_{m+1} \prod_{i=0}^{m-2} e_{t,i} + 3c_3 e_n\right). \tag{2.10}$$

Now, dividing (2.10) by (2.7) and the simplifying we get

$$\frac{H_m''(w_n)}{2g'(w_n)} \sim \left(c_2 - (-1)^{m-1} c_{m+1} \prod_{i=0}^{m-2} e_{t,i} + (3c_3 - 2c_2^2) e_n\right).$$

Therefore,

$$\alpha_n \sim \left(c_2 - (-1)^{m-1} c_{m+1} \prod_{i=0}^{m-2} e_{t,i} + (3c_3 - 2c_2^2)e_n\right),$$

and so,

$$\alpha_n - c_2 \sim \left(-(-1)^{m-1} c_{m+1} \prod_{i=0}^{m-2} e_{t,i} + (3c_3 - 2c_2^2)e_n\right). \qquad \square$$

Theorem 2.2. *If the errors of approximations $e_i = w_i - \gamma$ generated by an iterative technique satisfy:*

$$e_{k+1} \sim \prod_{i=0}^{m-2} (e_{k-i})^{m_i}, \quad k \geq k(e_k),$$

then the R-order of convergence of iterative technique, denoted with $O_R(\gamma)$, satisfies the inequality $O_R(\gamma) \geq q^*$, where q^* is the unique positive solution of the equation $q^{n+1} - \sum_{i=0}^{n} m_i q^{n-i} = 0$.

As a result, we arrive at the following conclusion on the convergence theorem for the iterative technique with memory (2.2).

Theorem 2.3. *Let α_n represent the variable in the iterative technique* (2.2)*, which is calculated by* (2.5)*. If an initial approximation w_0 is close enough to a simple root of $g(w)$, the iterative method* (2.2)–(2.5) *with memory has an R-order of convergence of at least* 5*.*

Proof. Initially, we will suppose that the R-order convergence of the sequences $\{w_n\}$ and $\{z_n\}$ is at least r and p. Hence, $e_{n+1} \sim E_{n,r} e_n^r$, where $E_{n,r}$ is an asymptotic error constant. The above relation may be also re-written as

$$e_{n+1} \sim E_{n,r}(E_{n-1,r} e_{n-1}^r)^r \sim E_{n,r} E_{n-1,r}^r e_{n-1}^{r^2} \tag{2.11}$$

and

$$e_{n,z} \sim E_{n,p} e_n^p,$$

or

$$e_{n,z} \sim E_{n,p}(E_{n-1,r} e_{n-1}^r)^p \sim E_{n,p} E_{n-1,r}^p e_{n-1}^{rp}. \tag{2.12}$$

By error expressions (2.3) and (2.4), it may be written as

$$e_{n,z} \sim z_n - \alpha \sim (-\alpha_n + c_2)e_n^2 + O(e_n^3), \tag{2.13}$$

$$e_{n+1} \sim D_{n,4}(\alpha_n - c_2)e_n^4 + O(e_n^5), \tag{2.14}$$

where $D_{n,4}$ is a varying quantity. Now, applying Theorem 2.1 for the case of $m = 4$, where $t_0 = z_{n-1}$, $t_1 = w_{n-1}$ and $t_2 = w_{n-1}$, we get

$$\alpha_n - c_2 \sim c_5 e_{t,0} e_{t,1} e_{t,2} = c_5 e_{n-1,z} e_{n-1}^2. \tag{2.15}$$

Substituting the relation (2.15) into the expressions (2.13) and (2.14), we obtain

$$
\begin{aligned}
e_{n,z} &\sim c_5 e_{n-1,z} e_{n-1}^2 e_n^2 \sim c_5 e_{n-1,z} e_{n-1}^2 (E_{n-1,r} e_{n-1}^r)^2 \sim c_5 e_{n-1,z} e_{n-1}^2 E_{n-1,r}^2 e_{n-1}^{2r} \\
&\sim c_5 (E_{n-1,p} e_{n-1}^p) e_{n-1}^2 E_{n-1,r}^2 e_{n-1}^{2r} \\
&\sim c_5 E_{n-1,r}^2 E_{n-1,p} e_{n-1}^{2r+p+2} \qquad (2.16)
\end{aligned}
$$

and

$$
\begin{aligned}
e_{n+1} &\sim D_{n,4} c_5 e_{n-1,z} e_{n-1}^2 e_n^4 \sim D_{n,4} c_5 (E_{n-1,p} e_{n-1}^p) e_{n-1}^2 (E_{n-1,r} e_{n-1}^r)^4 \\
&\sim D_{n,4} c_5 E_{n-1,p} E_{n-1,r}^4 e_{n-1}^{4r+p+2}. \qquad (2.17)
\end{aligned}
$$

By comparing the components of e_{n-1} in the two sets of relations (2.12)–(2.16) and (2.11)–(2.17), we arrive at the following system of equations:

$$
\begin{aligned}
2r + p + 2 &= rp, \\
4r + p + 2 &= r^2. \qquad (2.18)
\end{aligned}
$$

The positive solution to the system (2.18) is provided by the values $p = 3$ and $r = 5$. As a result, when α_n is determined by (2.5), the R-order of the method with memory (2.2) is reached to at least 5. $\square$

An alternative proof. The method discussed in reference [15], known as the Herzberger's matrix approach, is now being utilized on the order of single step s-point method $x_k = \Psi(x_{k-1}, x_{k-2}, \dots, x_{k-s})$. A matrix $A^{(s)} = (a_{ij})$, associated with this method, has the elements

$$
\begin{aligned}
a_{1,j} &= \text{amount of information required at point } x_{k-j}, \quad j = 1, 2, 3, \dots, s, \\
a_{i,i-1} &= 1, \quad i = 2, 3, \dots, s, \\
a_{i,j} &= 0, \quad \text{otherwise.} \qquad (2.19)
\end{aligned}
$$

The order of an s-step method $\Psi = \Psi_1 \circ \Psi_2 \circ \cdots \circ \Psi_s$ is the spectral radius of the product of matrices $A^{(s)} = A_1 \cdot A_2 \cdots A_s$. We may express each estimate w_{n+1}, z_n as a function of available information $g(z_n)$ and $g(w_n)$ from the n-th iteration and $g(z_{n-1})$ and $g(w_{n-1})$ from the previous iteration, depending on the accelerating technique. We construct the relevant matrices from the relations (2.2), (2.5) and (2.19) as follows:

$$
\begin{aligned}
w_{n+1} =& \Psi_1(z_n, w_n, z_{n-1}) \Rightarrow A_1 = \begin{bmatrix} 1 & 2 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \\
z_n =& \Psi_2(w_n, z_{n-1}, w_{n-1}) \Rightarrow A_2 = \begin{bmatrix} 2 & 1 & 2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.
\end{aligned}
$$

Thus, we acquire

$$A^{(2)} = A_1 \cdot A_2 = \begin{bmatrix} 4 & 1 & 2 \\ 2 & 1 & 2 \\ 1 & 0 & 0 \end{bmatrix},$$

whose eigenvalues are $(5, 0, 0)$ and spectral radius of the matrix $A^{(2)}$ is 5. Therefore, the order of convergence of with memory method (2.2) is five. □

Now, by making some more modification in the scheme (2.2) at this time, we are attempting to enhance its convergence order. Consider the following new updated version of the scheme (2.2), where an additional parameter β_n is added in the second sub-step, we get a new bi-parametric two-step iterative method with memory given by:

$$\begin{aligned} \alpha_n &= \frac{H_4{}''(w_n)}{2g'(w_n)}, \\ z_n &= w_n - \frac{g(w_n)}{g'(w_n) - \alpha_n g(w_n)}, \\ w_{n+1} &= z_n - \frac{g(z_n)}{2\left(\frac{g(z_n)-g(w_n)}{z_n-w_n}\right) - g'(w_n) - \beta_n g(w_n)^2}. \end{aligned} \tag{2.20}$$

It's error equation is:

$$e_{n+1} = (\alpha_n - c_2)(g'(\gamma)\beta_n + (\alpha_n - c_2)c_2 + c_3)e_n^4 + O(e_n^5), \tag{2.21}$$

where $e_n = w_n - \gamma$ and $c_i = \frac{g^{(i)}(\gamma)}{i! g'(\gamma)}$, for $i = 2, 3, 4, \ldots$, and $\gamma \in \mathbb{R}$. It is symbolised by **OWM6**. It is clear from (2.21) that the order of convergence of scheme (2.20) is five for $\beta_n \neq \frac{-c_3}{g'(\gamma)}$ and when $\beta_n = -\frac{c_3}{g'(\gamma)} = -\frac{g'''(\gamma)}{3! g'(\gamma)^2}$, the convergence order of method (2.20) would be higher. However, exact values of $g'(\gamma)$ and $g'''(\gamma)$ are not available for this type of convergence acceleration and so we can use the data available from the most recent iteration and the one before it, and it satisfies the condition $\lim_{n\to+\infty} \beta_n = -\frac{c_3}{g'(\gamma)} = -\frac{g'''(\gamma)}{3! g'(\gamma)^2}$ for the asymptotic constant to be zero in the relation (2.21). For the calculation of β_n, we consider the following best possible expression:

$$\beta_n = -\frac{H_5{}'''(z_n)}{3! g'(w_n)^2}, \tag{2.22}$$

where

$$\begin{aligned} H_5(w) =& g(z_n) + (w - z_n)g[z_n, w_n] + (w - z_n)(w - w_n)g[z_n, w_n, w_n] \\ &+ (w - z_n)(w - w_n)^2 g[z_n, w_n, w_n, z_{n-1}] + (w - z_n)(w - w_n)^2(w - z_{n-1}) \\ &\times g[z_n, w_n, w_n, z_{n-1}, w_{n-1}] + (w - z_n)(w - w_n)^2(w - z_{n-1})(w - w_{n-1}) \\ &\times g[z_n, w_n, w_n, z_{n-1}, w_{n-1}, w_{n-1}], \end{aligned}$$

and so,

$$\begin{aligned}H_5'''(z_n) =&6g[z_n, w_n, w_n, z_{n-1}] + (12(z_n - w_n) + 6(z_n - z_{n-1}))g[z_n, w_n, w_n, z_{n-1}, w_{n-1}]\\&+ \Big(6(z_n - w_n)^2 + 12(z_n - w_n)(z_n - z_{n-1}) + 12(z_n - w_n)(z_n - w_{n-1})\\&+ 6(z_n - z_{n-1})(z_n - w_{n-1})\Big)g[z_n, w_n, w_n, z_{n-1}, w_{n-1}, w_{n-1}].\end{aligned}$$

Theorem 2.4. *Let a Hermite interpolating polynomial H_m of degree m which interpolates a function g at nodes $z_n, w_n, w_n, t_0, \dots, t_{m-3}$ located within an interval I, and the derivative $g^{(m+1)}$ is continuous in I, as well as the Hermite interpolating polynomial satisfying the conditions $H_m(z_n) = g(z_n)$, $H_m{}'(z_n) = g'(z_n)$, $H_m(w_n) = g(w_n)$, $H_m{}'(w_n) = g'(w_n)$, $H_m(t_i) = f(t_i)$, $i = 0, 1, \dots, m-3$. Indicate the errors $e_{t,i} = t_i - \gamma$, $i = 0, 1, 2, \dots, m-3$, and presume that*

(1) *all nodes $z_n, w_n, t_0, \dots, t_{m-3}$ are adequately near to the zero γ;*
(2) *the condition $e_n = O(e_{t,0}, e_{t,1}, \dots, e_{t,m-3})$ and $e_{n,z} = z_n - \gamma = O\left(e_n^2, e_{t,0}, \dots, e_{t,m-3}\right)$ hold.*

Then

$$H_m{}'''(z_n) = 3!g'(\gamma)\left(c_3 - (-1)^{m-2}c_{m+1}\prod_{i=0}^{m-3} e_{t,i} + 4c_4e_{n,z}\right)$$

and

$$g'(\gamma)\beta_n + c_3 \sim \left(-(-1)^{m-2}c_{m+1}\prod_{i=0}^{m-3} e_{t,i}\right).$$

Proof. The error expression for Hermite interpolating polynomial can be written as

$$g(w) - H_m(w) = \frac{g^{(m+1)}(\xi)}{(m+1)!}(w - z_n)(w - w_n)^2\prod_{i=0}^{m-3}(w - t_i), \quad \xi \in I.$$

After differentiating the aforementioned expression thrice at the point $w = z_n$ will give

$$g'''(z_n) - H_m'''(z_n) = 3!\frac{g^{(m+1)}(\xi)}{(m+1)!}\prod_{i=0}^{m-3}(z_n - t_i), \quad \xi \in I,$$

or

$$H_m'''(z_n) = g'''(z_n) - 3!\frac{g^{(m+1)}(\xi)}{(m+1)!}\prod_{i=0}^{m-3}(z_n - t_i), \quad \xi \in I. \tag{2.23}$$

Using Taylor's series expansion for derivatives of g at the point $z_n \in I$ and $\xi \in I$ about the root γ of g gives

$$\begin{aligned}g'(z_n) =&g'(\gamma)\Big(1 + 2c_2e_{n,z} + 3c_3e_{n,z}^2 + O(e_{n,z}^3)\Big),\\g''(z_n) =&g'(\gamma)\Big(2c_2 + 3!c_3e_{n,z} + O(e_{n,z}^2)\Big),\\g'''(z_n) =&g'(\gamma)\Big(3!c_3 + 4!c_4e_{n,z} + O(e_{n,z}^2)\Big)\end{aligned} \tag{2.24}$$

and

$$g^{(m+1)}(\xi) = g'(\gamma)\Big((m+1)!c_{m+1} + (m+2)!c_{m+2}e_\xi + O(e_\xi^2)\Big). \tag{2.25}$$

Putting the expansions (2.24) and (2.25) in the relation (2.23), we get

$$H_m'''(z_n) = 3!g'(\gamma)\left(c_3 - (-1)^{m-2}c_{m+1}\prod_{i=0}^{m-3} e_{t,i} + 4c_4e_{n,z}\right). \tag{2.26}$$

Now, dividing the relation (2.26) by $g'(w_n)^2$, we obtain

$$-\frac{H_m'''(z_n)}{3!g'(w_n)^2} \sim -\frac{1}{g'(\gamma)}\left(c_3 - (-1)^{m-2}c_{m+1}\prod_{i=0}^{m-3} e_{t,i}\right)$$

or

$$\beta_n \sim -\frac{1}{g'(\gamma)}\left(c_3 - (-1)^{m-2}c_{m+1}\prod_{i=0}^{m-3} e_{t,i}\right),$$

and hence

$$g'(\gamma)\beta_n + c_3 \sim (-1)^{m-2}c_{m+1}\prod_{i=0}^{m-3} e_{t,i}.$$

Thus the proof is finished. □

Theorem 2.5. *Let β_n be the changing parameter in the iterative method* (2.20), *and be computed by* (2.22). *If an initial approximation w_0 is close enough to a simple root of $g(w)$, the iterative method* (2.20)–(2.22) *with memory has R-order of convergence of at least* 5.70156.

Proof. Initially, let us suppose that the R-order convergence of the sequences $\{w_n\}$ and $\{z_n\}$ is at least r and p. So, that

$$e_{n+1} \sim E_{n,r}e_n^r, \tag{2.27}$$

where $E_{n,r}$ is an asymptotic error constant. Now, the relation (2.27) may be also re-written as

$$e_{n+1} \sim E_{n,r}(E_{n-1,r}e_{n-1}^r)^r \sim E_{n,r}E_{n-1,r}^r e_{n-1}^{r^2} \tag{2.28}$$

and

$$\begin{aligned} e_{n,z} &\sim E_{n,p}e_n^p, \\ e_{n,z} &\sim E_{n,p}(E_{n-1,r}e_{n-1}^r)^p \sim E_{n,p}E_{n-1,r}^p e_{n-1}^{rp}. \end{aligned} \tag{2.29}$$

By error expressions (2.3) and (2.21), it may be written as

$$e_{n,z} \sim z_n - \gamma \sim (-\alpha_n + c_2)e_n^2 + O(e_n^3), \tag{2.30}$$

$$e_{n+1} \sim (\alpha_n - c_2)\Big(g'(\gamma)\beta_n + (\alpha_n - c_2)c_2 + c_3\Big)e_n^4 + O(e_n^5). \tag{2.31}$$

Using Theorem 2.4 for $m = 5$, $t_0 = z_{n-1}$, $t_1 = w_{n-1}$ and $t_2 = w_{n-1}$, we obtain

$$g'(\gamma)\beta_n + c_3 \sim c_6e_{t,0}e_{t,1}e_{t,2} = c_6e_{n-1,z}e_{n-1}^2. \tag{2.32}$$

Using the relations (2.15) into (2.30) and (2.32) in the expression (2.31), it can be written as

$$
\begin{aligned}
e_{n,z} &\sim c_5 e_{n-1,z} e_{n-1}^2 e_n^2 \sim c_5 e_{n-1,z} e_{n-1}^2 (E_{n-1,r} e_{n-1}^r)^2, \\
&\sim c_5 e_{n-1,z} e_{n-1}^2 E_{n-1,r}^2 e_{n-1}^{2r} \sim c_5 (E_{n-1,p} e_{n-1}^p) e_{n-1}^2 E_{n-1,r}^2 e_{n-1}^{2r} \\
&\sim c_5 E_{n-1,r}^2 E_{n-1,p} e_{n-1}^{2r+p+2} \qquad (2.33)
\end{aligned}
$$

and

$$
\begin{aligned}
e_{n+1} &\sim (c_5 e_{n-1,z} e_{n-1}^2)\Big(c_6 e_{n-1,z} e_{n-1}^2 + c_2 c_5 e_{n-1,z} e_{n-1}^2\Big) e_n^4 \\
&\sim c_5 (c_6 + c_2 c_5) e_{n-1,z}^2 e_{n-1}^4 e_n^4 \sim c_5 (c_6 + c_2 c_5)(E_{n-1,p} e_{n-1}^p)^2 e_{n-1}^4 (E_{n-1,r} e_{n-1}^r)^4 \\
&\sim c_5 (c_6 + c_2 c_5) E_{n-1,p}^2 E_{n-1,r}^4 e_{n-1}^{4r+2p+4}. \qquad (2.34)
\end{aligned}
$$

By comparing the components of e_{n+1} in (2.34)–(2.28) and (2.33)–(2.29), we arrive at the following system of equations:

$$
\begin{aligned}
4r + 2p + 4 &= r^2, \\
2r + p + 2 &= rp. \qquad (2.35)
\end{aligned}
$$

The positive solution to the system (2.35) is provided by $p = 2.85078$, $r = 5.70156$. As a result, when β_n is determined by formula (2.22), the R-order of the method with memory scheme (2.20) is at least 5.70156. □

An alternative proof. From the relations (2.20), (2.22) and similar to that used in the alternative proof of the previous Theorem 2.3, we derive the corresponding matrices:

$$
\begin{aligned}
w_{n+1} =& \Psi_1(z_n, w_n, z_{n-1}, w_{n-1}) \Rightarrow A_1 = \begin{bmatrix} 1 & 2 & 1 & 2 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \\
z_n =& \Psi_2(w_n, z_{n-1}, w_{n-1}, z_{n-2}) \Rightarrow A_2 = \begin{bmatrix} 2 & 1 & 2 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.
\end{aligned}
$$

Thus, we acquire

$$
A^{(2)} = A_1 \cdot A_2 = \begin{bmatrix} 4 & 2 & 4 & 0 \\ 2 & 1 & 2 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},
$$

whose eigenvalues are $(5.70156, -0.701562, 0, 0)$ and spectral radius of the matrix $A^{(2)}$ is 5.70156. Therefore, the order of convergence of with memory method (2.20) is 5.70156. □

3. Numerical Results and Comparisons

In this part, we will numerically compare the considered uni-parametric two-step with memory scheme $OWM4$ along with the similar nature schemes $XW(16-18)$, $XW(16-19)$, $XW(16-20)$, $XW(17-18)$, $XW(17-19)$ and $XW(17-20)$ considered in [16] and $NC4(2.4-2.5)$, $NC4(2.4-2.6)$ and $NC4(2.4-2.7)$ presented in the article [17] and presented bi-parametric with memory method $OWM6$ along with the proposed in [18]. Wang [18] presented two bi-parametric iterative methods with memory as mentioned below:

$$\begin{aligned} z_n &= w_n - \frac{g(w_n)}{g'(w_n) - \alpha_n g(w_n)}, \\ w_{n+1} &= z_n - \frac{g(z_n)}{2g[w_n, z_n] - g'(w_n)} \cdot \left(\frac{g'(w_n)^2}{g'(w_n)^2 - \beta_n g(w_n)^2} \right), \end{aligned} \tag{3.1}$$

which is represented by the symbol **XW1** and

$$\begin{aligned} z_n &= w_n - \frac{g(w_n)}{g'(w_n) - \alpha_n g(w_n)}, \\ w_{n+1} &= z_n - \frac{g(z_n)}{2g[w_n, z_n] - g'(w_n)} \left(1 + \beta_n \left(\frac{g(w_n)}{g'(w_n)} \right)^2 \right), \end{aligned} \tag{3.2}$$

which is denoted by **XW2**. In the following form, they have captured the values of the two parameters α_n and β_n for both methods.

Method 1:

$$\alpha_n = \frac{H_4''(w_n)}{2g'(w_n)} \quad \text{and} \quad \beta_n = -\frac{H_4'''(w_n)}{6g'(w_n)}, \tag{3.3}$$

where

$$H_4'''(w_n) = 6g[w_n, w_n, z_{n-1}, w_{n-1}] + 6(2w_n - w_{n-1} - z_{n-1})g[w_n, z_n, z_{n-1}, w_{n-1}, w_{n-1}].$$

Method 2:

$$\alpha_n = \frac{H_4''(w_n)}{2g'(w_n)} \quad \text{and} \quad \beta_n = -\frac{H_3'''(z_n)}{6g'(w_n)}, \tag{3.4}$$

where $H_3'''(z_n) = 6g[z_n, w_n, w_n, z_{n-1}]$.

Method 3:

$$\alpha_n = \frac{H_4''(w_n)}{2g'(w_n)} \quad \text{and} \quad \beta_n = -\frac{H_4'''(z_n)}{6g'(w_n)}, \tag{3.5}$$

where $H_4'''(z_n) = H_3'''(z_n) + 6(3z_n - z_{n-1} - 2w_n)g[z_n, w_n, w_n, z_{n-1}, w_{n-1}]$.

Method 4:

$$\alpha_n = \frac{H_4''(w_n)}{2g'(w_n)} \quad \text{and} \quad \beta_n = -\frac{H_5'''(z_n)}{6g'(w_n)}, \tag{3.6}$$

where $H_5'''(z_n) = H_4'''(z_n) + 6[(z_n - z_{n-1})(z_n - w_{n-1}) + (z_n - w_n)^2 + 2(z_n - w_n)(2z_n - z_{n-1} - w_{n-1})]g[z_n, w_n, w_n, z_{n-1}, w_{n-1}]$.

Table 1. Test functions and their roots.

Nonlinear function	Root
$g_1 = we^{w^2} - \sin^2 w + 3\cos w + 5$	-1.2676...
$g_2 = w^5 + w^4 + 4w^2 - 15$	1.3474...
$g_3 = w^3 - w^2 - 1$	1.4655...

Table 1 includes the roots of three nonlinear test functions (taken from [18, 19]). The numerical results shown in Table 2 and 3 are consistent with the theory presented in this discussion. The absolute errors $|w_n - \gamma|$ upto three iterate have been calculated. For numerical computation MATHEMATICA 8 is used. Now, according to Weerakoon [20], the formula below can be used to estimate the computational order of convergence,

$$COC \approx \frac{\log |g(w_{n+1})/g(w_n)|}{\log |g(w_n)/g(w_{n-1})|},$$

to verify the established theoretical rate of convergence. Table 2 and 3 confirms the significance of the presented with memory scheme over some well published similar nature algorithms.

4. Conclusion

In this article, we have presented a two-step with memory iterative method for finding the solution of nonlinear equations. Because our goal is to develop the method of higher-order convergence without any extra functional computations. To obtain higher-order convergence without any extra computations, we have employed one and two self-accelerating parameters that are constructed by Hermite interpolating polynomials in the well-established optimal fourth-order without memory scheme. The order of convergence for the new suggested two-step iterative with memory has risen from 4 to 5.70156, which is also verified by an alternate approach called Herzberger's matrix method. The numerical results have been provided to validate the theoretical outcomes.

Acknowledgements. The authors are sincerely thankful to the reviewers for their valuable feedback and expertise in reviewing our research paper, their contributions have greatly improved the quality of our work. We are also thankful to the Department of Science and Technology, New Delhi, India for approving the proposal under the scheme FIST program (Ref. No. SR/FST/MS/2022 dated 19.12.2022).

TABLE 2. Numerical comparison of single parametric two-point with memory method

Method	$\lvert w_1-\gamma\rvert$	$\lvert w_2-\gamma\rvert$	$\lvert w_3-\gamma\rvert$	COC
Example g_1, initial guess $w_0=-1.6$				
XW $(16)-(18)$, $\alpha_0=-0.01$	$3.5037e-2$	$2.8246e-6$	$1.2080e-25$	4.7042
XW $(16)-(19)$, $\alpha_0=-0.01$	$3.5037e-2$	$4.7605e-7$	$1.7515e-27$	4.1782
XW $(16)-(20)$, $\alpha_0=-0.01$	$3.5037e-2$	$3.4949e-7$	$4.1255e-28$	4.1649
XW $(17)-(18)$, $\alpha_0=-0.01$	$1.8398e-2$	$2.1773e-7$	$1.3052e-30$	4.7018
XW $(17)-(19)$, $\alpha_0=-0.01$	$1.8398e-2$	$3.0276e-8$	$1.7117e-32$	4.1837
XW $(17)-(20)$, $\alpha_0=-0.01$	$1.8398e-2$	$3.5032e-8$	$2.7935e-32$	4.2025
NC $(2.4)-(2.5)$, $\alpha_0=-0.01$	$1.8880e-2$	$2.3820e-7$	$1.9513e-30$	4.7005
NC $(2.4)-(2.6)$, $\alpha_0=-0.01$	$1.8880e-2$	$3.3604e-8$	$2.6359e-32$	4.1835
NC $(2.4)-(2.7)$, $\alpha_0=-0.01$	$1.8880e-2$	$3.8273e-8$	$4.0253e-32$	4.2025
OWM4 $(2.4)-(2.7)$, $\alpha_0=-0.01$	$1.8309e-2$	$2.9275e-8$	$5.2296e-38$	5.1217
Example g_2, initial guess $w_0=1.4$				
XW $(16)-(18)$, $\alpha_0=-0.01$	$1.8371e-5$	$1.3038e-22$	$7.0315e-101$	4.5640
XW $(16)-(19)$, $\alpha_0=-0.01$	$1.8371e-5$	$1.5797e-22$	$5.6795e-107$	4.3243
XW $(16)-(20)$, $\alpha_0=-0.01$	$1.8371e-5$	$6.7085e-25$	$1.5685e-108$	4.3026
XW $(17)-(18)$, $\alpha_0=-0.01$	$3.8040e-6$	$2.3630e-25$	$3.2074e-113$	4.5748
XW $(17)-(19)$, $\alpha_0=-0.01$	$3.8040e-6$	$4.3246e-27$	$9.8591e-118$	4.3278
XW $(17)-(20)$, $\alpha_0=-0.01$	$3.8040e-6$	$2.2214e-28$	$7.0328e-124$	4.2953
NC $(2.4)-(2.5)$, $\alpha_0=-0.01$	$3.7144e-6$	$2.1871e-25$	$2.2845e-113$	4.5752
NC $(2.4)-(2.6)$, $\alpha_0=-0.01$	$3.7144e-6$	$3.9924e-27$	$7.0907e-118$	4.3279
NC $(2.4)-(2.7)$, $\alpha_0=-0.01$	$3.7144e-6$	$1.9614e-28$	$4.0581e-124$	4.2951
OWM4 $(2.4)-(2.7)$, $\alpha_0=-0.01$	$3.8734e-6$	$2.4314e-28$	$2.8924e-139$	4.9961
Example g_3, initial guess $w_0=1.3$				
XW $(16)-(18)$, $\alpha_0=-0.01$	$1.5319e-2$	$4.8667e-10$	$2.6217e-44$	4.5743
XW $(16)-(19)$, $\alpha_0=-0.01$	$1.5319e-2$	$1.7723e-10$	$1.6516e-45$	4.4174
XW $(16)-(20)$, $\alpha_0=-0.01$	$1.5319e-2$	$1.7723e-10$	$3.3034e-45$	4.3794
XW $(17)-(18)$, $\alpha_0=-0.01$	$7.1877e-4$	$7.3695e-16$	$1.0978e-70$	4.5729
XW $(17)-(19)$, $\alpha_0=-0.01$	$7.1877e-4$	$3.5313e-17$	$5.5819e-75$	4.3430
XW $(17)-(20)$, $\alpha_0=-0.01$	$7.1877e-4$	$3.5313e-17$	$1.1164e-74$	4.3204
NC $(2.4)-(2.5)$, $\alpha_0=-0.01$	$7.1305e-4$	$7.3404e-16$	$1.0912e-70$	4.5737
NC $(2.4)-(2.6)$, $\alpha_0=-0.01$	$7.1305e-4$	$3.3934e-17$	$4.6559e-75$	4.3431
NC $(2.4)-(2.7)$, $\alpha_0=-0.01$	$7.1305e-4$	$3.3934e-17$	$9.3119e-75$	4.3205
OWM4 $(2.4)-(2.7)$, $\alpha_0=-0.01$	$0.7357e-4$	$3.9816e-17$	$1.8529e-83$	4.9998

TABLE 3. Numerical comparison of bi-parametric two-point with memory scheme

Method	$\lvert w_1 - \gamma \rvert$	$\lvert w_2 - \gamma \rvert$	$\lvert w_3 - \gamma \rvert$	COC
Example g_1, initial guess $w_0 = -1.5$				
XW1 (3.1) − (3.3), $\alpha_0 = \beta_0 = 0.01$	$0.527e-2$	$1.5696e-11$	$2.0984e-58$	5.4952
XW1 (3.1) − (3.4), $\alpha_0 = \beta_0 = 0.01$	$0.527e-2$	$4.1038e-12$	$3.1195e-62$	5.5000
XW1 (3.1) − (3.5), $\alpha_0 = \beta_0 = 0.01$	$0.527e-2$	$6.2388e-12$	$8.6664e-64$	5.8067
XW1 (3.1) − (3.6), $\alpha_0 = \beta_0 = 0.01$	$0.527e-2$	$2.6379e-12$	$1.5543e-68$	6.0433
XW2 (3.2) − (3.3), $\alpha_0 = \beta_0 = 0.01$	$0.527e-2$	$1.5694e-11$	$2.0977e-58$	5.4952
XW2 (3.2) − (3.4), $\alpha_0 = \beta_0 = 0.01$	$0.527e-2$	$4.1074e-12$	$3.1303e-62$	5.5002
XW2 (3.2) − (3.5), $\alpha_0 = \beta_0 = 0.01$	$0.527e-2$	$6.2364e-12$	$8.6536e-64$	5.8067
XW2 (3.2) − (3.6), $\alpha_0 = \beta_0 = 0.01$	$0.527e-2$	$2.6351e-12$	$1.5360e-68$	6.0435
OWM6 (2.30) − (2.32), $\alpha_0 = \beta_0 = 0.01$	$0.957e-2$	$1.086e-12$	$3.0103e-87$	5.9901
Example g_2, initial guess $w_0 = 1.5$				
XW1 (3.1) − (3.3), $\alpha_0 = \beta_0 = 0.01$	$0.2261e-3$	$3.1146e-22$	$9.2151e-116$	5.2365
XW1 (3.1) − (3.4), $\alpha_0 = \beta_0 = 0.01$	$0.2261e-3$	$1.0210e-21$	$7.3405e-117$	5.4852
XW1 (3.1) − (3.5), $\alpha_0 = \beta_0 = 0.01$	$0.2261e-3$	$1.7755e-22$	$2.1424e-124$	5.6292
XW1 (3.1) − (3.6), $\alpha_0 = \beta_0 = 0.01$	$0.2261e-3$	$1.5437e-22$	$2.7763e-128$	5.8211
XW2 (3.2) − (3.3), $\alpha_0 = \beta_0 = 0.01$	$0.2261e-3$	$3.1146e-22$	$9.2152e-116$	5.2365
XW2 (3.2) − (3.4), $\alpha_0 = \beta_0 = 0.01$	$0.2261e-3$	$1.0211e-21$	$7.3412e-117$	5.4852
XW2 (3.2) − (3.5), $\alpha_0 = \beta_0 = 0.01$	$0.2261e-3$	$1.7755e-22$	$2.1423e-124$	5.6292
XW2 (3.2) − (3.6), $\alpha_0 = \beta_0 = 0.01$	$0.2261e-3$	$1.5436e-22$	$2.7762e-128$	5.8210
OWM6 (2.30) − (2.32), $\alpha_0 = \beta_0 = 0.01$	$0.1794e-4$	$1.8535e-29$	$4.8802e-166$	5.6942
Example g_3, initial guess $w_0 = 1.3$				
XW1 (3.1) − (3.3), $\alpha_0 = \beta_0 = 0.2$	$0.1813e-3$	$1.0922e-23$	$5.2152e-139$	6.0000
XW1 (3.1) − (3.4), $\alpha_0 = \beta_0 = 0.2$	$0.1813e-3$	$1.0922e-23$	$5.2152e-139$	6.0000
XW1 (3.1) − (3.5), $\alpha_0 = \beta_0 = 0.2$	$0.1813e-3$	$1.0922e-23$	$5.2152e-139$	6.0000
XW1 (3.1) − (3.6), $\alpha_0 = \beta_0 = 0.2$	$0.1813e-3$	$1.0922e-23$	$5.2152e-139$	6.0000
XW2 (3.2) − (3.3), $\alpha_0 = \beta_0 = 0.2$	$0.1830e-3$	$1.1536e-23$	$7.2406e-139$	5.9999
XW2 (3.2) − (3.4), $\alpha_0 = \beta_0 = 0.2$	$0.1830e-3$	$1.1536e-23$	$7.2406e-139$	5.9999
XW2 (3.2) − (3.5), $\alpha_0 = \beta_0 = 0.2$	$0.1830e-3$	$1.1536e-23$	$7.2406e-139$	5.9999
XW2 (3.2) − (3.6), $\alpha_0 = \beta_0 = 0.2$	$0.1830e-3$	$1.1536e-23$	$7.2406e-139$	5.9999
OWM6 (2.30) − (2.32), $\alpha_0 = \beta_0 = 0.2$	$0.4008e-4$	$2.114e-28$	$4.5555e-168$	6.0000

References

[1] A. Cordero, J. R. Torregrosa and M. P. Vassileva, *A family of modified Ostrowski's methods with optimal eighth order of convergence*, Appl. Math. Lett. **24**(12) (2011), 2082–2086. https://doi.org/10.1016/j.aml.2011.06.002

[2] A. Cordero, T. Lotfi, K. Mahdiani and J. R. Torregrosa, *Two optimal general classes of iterative methods with eighth-order*, Acta Appl. Math. **134**(1) (2014), 61–74. https://doi.org/10.1007/s10440-014-9869-0

[3] J. P. Jaiswal, *Some class of third-and fourth-order iterative methods for solving nonlinear equations*, J. Appl. Math. **2014**(1) (2014), 1–17. https://doi.org/10.1155/2014/817656

[4] T. Lotfi, F. Soleymani, M. Ghorbanzadeh and P. Assari, *On the construction of some tri-parametric iterative methods with memory*, Numer. Algorithms **70** (2015), 835–845. https://doi.org/10.1007/s11075-015-9976-7

[5] S. Kumar, V. Kanwar, S. K. Tomar and S. Singh, *Geometrically constructed families of Newton's method for unconstrained optimization and nonlinear equations*, Int. J. Math. Math. Sci. **2011** (2011), 1–9. https://doi.org/10.1155/2011/972537

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

[7] M. Petković, B. Neta, L. Petković and J. Džunić, *Multipoint Methods for Solving Nonlinear Equations*, Academic Press, New York, 2012.

[8] J. Džunić and M. S. Petković, *On generalized multipoint root-solvers with memory*, J. Comput. Appl. Math. **236**(11) (2012), 2909–2920. https://doi.org/10.1016/j.cam.2012.01.035

[9] J. Džunić, *Modified Newton's method with memory*, Facta Univ. Ser. Math. Inform. **28**(4) (2013), 429–441.

[10] J. Džunić and M. S. Petković, *On generalized biparametric multipoint root finding methods with memory*, J. Comput. Appl. Math. **255** (2014), 362–375. https://doi.org/10.1016/j.cam.2013.05.013

[11] T. Lotfi and P. Assari, *New three-and four-parametric iterative with memory methods with efficiency index near* 2, Appl. Math. Comput. **270** (2015), 1004–1010. https://doi.org/10.1016/j.amc.2015.08.017

[12] J. P. Jaiswal, *Improved bi-accelerator derivative free with memory family for solving nonlinear equations*, J. Appl. Anal. Comput. **6**(1) (2016), 196–206. https://doi.org/10.11948/2016016

[13] J. Džunić, L. D. Petković and M. S. Petković, *On an application of Herzberger's matrix method to multipoint families of root-solvers*, Filomat **32**(11) (2018), 3815–3829. https://doi.org/10.2298/FIL1811815D

[14] S. K. Khattri, *Quadrature based optimal iterative methods with applications in high-precision computing*, Numer. Math. Theory Methods Appl. **5**(4) (2012), 592–601. https://doi.org/10.4208/nmtma.2012.m1114

[15] J. Herzberger, *Über matrixdarstellungen für iterationverfahren bei nichtlinearen gleichungen*, Computing **12**(3) (1974), 215–222.

[16] X. Wang and T. Zhang, *A new family of Newton-type iterative methods with and without memory for solving nonlinear equations*, Calcolo **51**(1) (2014), 1–15. https://doi.org/10.1007/s10092-012-0072-2

[17] N. Choubey and J. P. Jaiswal, *Two-and three-point with memory methods for solving nonlinear equations*, Numer. Anal. Appl. **10**(1) (2017), 74–89. https://doi.org/10.1134/S1995423917010086

[18] X. Wang and T. Zhang, *High-order Newton-type iterative methods with memory for solving nonlinear equations*, Math. Commun. **19**(1) (2014), 91–109.

[19] N. Choubey, B. Panday and J. P. Jaiswal, *Several two-point with memory iterative methods for solving nonlinear equations*, Afr. Mat. **29** (2018), 435–449. https://doi.org/10.1007/s13370-018-0552-x

[20] S. Weerakoon and T. Fernando, *A variant of Newton's method with accelerated third-order convergence*, Appl. Math. Lett. **13**(8) (2000), 87–93. `https://doi.org/10.1016/S0893-9659(00)00100-2`

[1,2]Department of Mathematics,
Guru Ghasidas Vishwavidyalaya (A Central University),
Bilaspur 495009, Chhattisgarh, India.
Email address: `nishantyd001@gmail.com`
Email address: `asstprofjpmanit@gmail.com`