Int. J. Appl. Comput. Math (2022) 8:83
https://doi.org/10.1007/s40819-022-01283-8

ORIGINAL PAPER

Family of Multipoint with Memory Iterative Schemes for Solving Nonlinear Equations

Neha Choubey[1] · **J. P. Jaiswal**[2] · **Abhishek Choubey**[3]

Accepted: 20 February 2022 / Published online: 23 March 2022

Abstract

In this article, we proposed a two-step with memory iterative scheme for solving nonlinear equations. We improved the convergence order of the existing fifth-order without memory iterative scheme by converting it into with memory method. The acceleration of convergence order is attained by using different possible approximations of self-accelerating parameters. These parameters are calculated using Hermite interpolating polynomial to improve the convergence rate of without memory scheme. This acceleration of convergence rate is attained without any extra functional evaluation. In particular, the R-order convergence of proposed two-step with memory scheme is accelerated from 5 to 5.8284, 6.0275 and 6.2128 without any new functions evaluation. An extra benefit of this technique is that the limitation $f'(x) \neq 0$, in the neighbourhood of the requisite root imposed on Newton's scheme is removed. The numerical evaluation is specified in the paper, to confirm the theoretical results.

Keywords Iterative scheme · Computational efficiency · With memory scheme · Numerical result

Mathematics Subject Classification 65B99 · 65H05 · 65G99

✉ J. P. Jaiswal
asstprofjpmanit@gmail.com

Neha Choubey
nehachby2@gmail.com; neha.choubey@vitbhopal.ac.in

Abhishek Choubey
abhishekchby2@gmail.com

[1] School of Advanced Sciences and Languages, VIT Bhopal University, Bhopal, M.P. 466114, India

[2] Department of Mathematics, Guru Ghasidas Vishwavidyalaya (A Central University), Bilaspur, C.G. 495009, India

[3] Department of Mechanical Engineering, Oriental Institute of Science and Technology, Bhopal, M.P. 462021, India

Introduction

The main goal of the article is to calculate the simple root of nonlinear equation $f(x)=0$ as it is a very important problem in real world phenomena, which attracted so much attention recently. In order to find a solution to these types of nonlinear equations, numerous multipoint iterative schemes have been proposed (one can see [1–18]). These iterative techniques play a vital role in area of research in numerical analysis because they have numerous applications in different branches of sciences, economics, dynamical model, engineering, and so on. Out of them the most eminent one-point iterative scheme without memory is the classical Newton–Raphson method, defined as follows:

$$z_{n+1}=z_n-\frac{f(z_n)}{f'(z_n)}, \tag{1}$$

used to find the solution of the nonlinear equation and it's convergence order is 2. A drawback of this scheme is the assumption $f'(z_n)\neq 0$, which limits its practical applications. To resolve this issue, Kumar et al. [12] developed a one-point iterative scheme, given as:

$$z_{n+1}=z_n-\frac{f(z_n)}{f'(z_n)-\gamma f(z_n)}. \tag{2}$$

It is clear from the above equation that by taking the value of $\gamma=0$ in (2), we will get Newton's method (1). The error equation for Kumar's approach is:

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

where $e_n=x_n-\alpha$, $c_i=\frac{1}{i!}\frac{f^{(i)}(\alpha)}{f'(\alpha)}$, $i=2,3,\ldots$ and α is the root of $f(z)=0$. The error equation given in (3) clearly shows that by taking $\gamma=c_2$, the convergence order of the method can be increased.

Next, we discuss the classification of possible type of iteration function (I.F.). These I.F. have been categorized on the basis of the information which they require [19, 20]:
(i) One-point iterative function without memory $x_{k+1}=\phi(x_k)$ can be determined by only new data at x_k, where ϕ is called a one-point I.F. The best known example of one-point I.F. type is Newton's I.F. On the other hand multi-point iteration function without memory $x_{k+1}=\phi(x_k,w_1(x_k),\ldots,w_n(x_k))$, can be determined by only new information at $x_k,w_1(x_k),\ldots,w_n(x_k)(n\geq 1)$, where ϕ is called multipoint I.F. without memory.

(ii) One-point iteration function with memory $x_{k+1}=\phi(x_k;x_{k-1},\ldots,x_{k-n})$ can be determined by new data at x_k and reused data at $x_{k-1},\ldots,x_{k-n}$, where ϕ is known as one-point I.F. with memory. The best known example of with memory one point iterative method is Secant method. On the other hand multi-point iteration function with memory $x_{k+1}=\phi(z_k;z_{k-1},\ldots,z_{k-n})$, where the I.F. ϕ which has arguments z_j, and each such argument represents $k+1$ quantities $x_j,w_1(x_j),\ldots,w_n(x_j)$
$(n\geq 1)$ is called a multipoint I.F. with memory. In the above mentioned mapping semicolon separates the points at which new information are used from the point at which old information are reused i.e. in each iterative step we must preserve information of the last n approximations x_j, and for each approximation, we must calculate n expressions $w_1(x_j),\ldots,w_n(x_j)$.

Multipoint iterative schemes are of great practical importance, since they overcome the theoretical limits of any one-point iterative scheme. Concerning their convergence order and computational efficiency. They also generate estimations of greater accuracy, highly improved computer arithmetic and symbolic calculation has allowed efficient execution of multipoint method. Memory-based multipoint iterative schemes use data from the current and previous iterations. Whereas the initial scheme for the construction of this type of scheme dates back to 1964 and Traub's book, contribution in this field appears very rarely in the literature. To fill this gap this article presents the with memory two-step scheme and convergence order of this scheme is greater than corresponding optimal without memory scheme. The improved order of convergence has been achieved without any extra evaluation of functions using several self-accelerating parameters, which results in higher computational efficiency.

The main objective of this article is to construct such types of methods whose computational efficiency is high. These schemes are referred as multipoint iterative schemes with memory, and they can accelerate the convergence order without memory schemes without any additional functional evaluation. The summary of the paper is as follows: In Sect. 2 the new with memory multipoint iterative scheme has been constructed by adding the self-accelerating parameter in the first step of the existing fifth order without memory method and the convergence analysis have been studied. To calculate the self-accelerating parameters we have used the Hermite interpolating polynomial. Using these parameter, the R-order convergence of the proposed scheme is accelerated from 5 to 5.8284, 6.0275 and 6.2128. Numerical calculation for several examples are performed in Sect. 3 to establish theoretical analysis. The concluding remarks are given in the last section.

Analysis of Convergence for with Memory Methods

In the following section, we will use the parameter T in the scheme proposed by Noor et al. [3] to increase its convergence by replacing the parameter T with iterative parameter T_n. For this first we have included parameter T in the first sub-step of without memory scheme of the fifth order, presented in the article [3]:

$$\begin{aligned} w_n &= z_n - \frac{f(z_n)}{f'(z_n) - Tf(z_n)}, \\ z_{n+1} &= w_n - \frac{2f(z_n)f(w_n)f'(w_n)}{2f(z_n)f'(w_n)^2 - f'(z_n)^2 f(w_n) + f'(z_n)f'(w_n)f(w_n)}. \end{aligned} \tag{4}$$

The error expressions of each sub-step of (4) are:

$$\begin{aligned} e_{n,w} = w_n - \alpha &= (c_2 - T)e_n^2 + (-2c_2^2 - T^2 + 2c_2T + 2c_3)e_n^3 \\ &\quad + (T^3 + 5Tc_2^2 - 4c_2^3 - 4Tc_3 + c_2(7c_3 - 3T^2) - 3c_4)e_n^4 + O(e_n^5), \end{aligned} \tag{5}$$

and

$$\begin{aligned} e_{n+1} = &\frac{-1}{2}(T - c_2)^2(2Tc_2 + 3c_3)e_n^5 \\ &-\frac{1}{2}(T - c_2)[2c_2^4 + 6c_2T^3 + c_2^2(6Tc_2 + 13c_3) + 2T(3T^2c_2 + 5Tc_3 + 2c_4) \\ &-(10T^2c_2^2 + 12c_3^2 + c_2(25Tc_3 + 4c_4))]e_n^6 + O(e_n^7), \end{aligned} \tag{6}$$

where $e_{n,w} = w_n - \alpha$, $e_n = z_n - \alpha$ and $c_i = \frac{f^{(i)}(\alpha)}{i!f'(\alpha)}$, for $i = 2, 3, \ldots$ and $T \in R$. We obtain the following with memory iterative scheme by replacing T with T_n in (4):

$$\begin{aligned} w_n &= z_n - \frac{f(z_n)}{f'(z_n) - T_n f(z_n)}, \\ z_{n+1} &= w_n - \frac{2f(z_n)f(w_n)f'(w_n)}{2f(z_n)f'(w_n)^2 - f'(z_n)^2 f(w_n) + f'({}_n)f'(w_n)f(w_n)}, \end{aligned} \tag{7}$$

and the above scheme is represented by MWM5. Now from Eq. (6) it is clear that the convergence order of algorithm (4) is five when $T \neq c_2$. Next to accelerate the order of convergence of the method (4) from five to higher, we can assume $T = c_2 = f''(\alpha)/(2f'(\alpha))$ but in actual fact the exact values of $f'(\alpha)$ and $f''(\alpha)$ are not attainable in practice. So we will assume the parameter T as T_n. The parameter T_n can be calculated by using the available data from the previous and current iterations and satisfies the condition $\lim_{n\to\infty} T_n = c_2 = f''(\alpha)/(2f'(\alpha))$, such that the fifth and sixth order asymptotic convergence constant should be zero in the error expression (6). Different formulas for T_n are as follows:
$Method$ 1:

$$T_n = \frac{H_2''(z_n)}{2f'(z_n)}, \tag{8}$$

where $H_2(z) = f(z_n) + f[z_n, z_n](z - z_n) + f[z_n, z_n, w_{n-1}](z - z_n)^2$ and $H_2''(z_n) = 2f[z_n, z_n, w_{n-1}]$.
$Method$ 2:

$$T_n = \frac{H_3''(z_n)}{2f'(z_n)}, \tag{9}$$

where $H_3(z) = H_2(z) + f[z_n, z_n, w_{n-1}, z_{n-1}](z - z_n)^2(z - w_{n-1})$ and $H_3''(z_n) = 2f[z_n, z_n, w_{n-1}] + 2f[z_n, z_n, w_{n-1}, z_{n-1}](z_n - w_{n-1})$.
$Method$ 3:

$$T_n = \frac{H_4''(z_n)}{2f'(z_n)}, \tag{10}$$

where $H_4(z) = H_3(z) + f[z_n, z_n, w_{n-1}, z_{n-1}, z_{n-1}](z - z_n)^2(z - w_{n-1})(z - z_{n-1})$ and $H_4''(z_n) = 2f[z_n, z_n, w_{n-1}] + (4f[z_n, z_n, w_{n-1}, z_{n-1}] - 2f[z_n, w_{n-1}z_{n-1}, z_{n-1}])(z_n - w_{n-1})$.
$Note$: The condition $H_m'(z_n) = f'(z_n)$ is satisfied by the Hermite interpolation polynomial $H_m(z)$ for $m = 2, 3, 4$. So, $T_n = \frac{H_m''(z_n)}{2f'(z_n)}$ can be expressed as $T_n = \frac{H_m''(z_n)}{2H_m'(z_n)}$ $(m = 2, 3, 4)$.

Theorem 1 *Let H_m be the Hermite polynomial of degree m that interpolates a function f at interpolation nodes $z_n, z_n, t_0 \dots t_{m-2}$ belonged in an interval I and the derivative $f^{(m+1)}$ is continuous in I and the Hermite polynomial $H_m(z_n) = f(z_n)$, $H_m'(z_n) = f'(z_n)$, $H_m(t_j) = f(t_j)$ $(j = 0, 1, \dots, m-2)$. Denote $e_{t,j} = t_j - \alpha$ $(j = 0, 1, \dots, m-2)$ and suppose that*

(1) all nodes $z_n, t_0, \dots, t_{m-2}$ are enough near to the root α:
(2) the condition $e_n = O(e_{t,0} \dots e_{t,m-2})$ holds. Then

$$H_m''(z_n) = 2f'(\alpha)\Big(c_2 - (-1)^{m-1} c_{m+1} \prod_{j=0}^{m-2} e_{t,j} + 3c_3 e_n\Big), \tag{11}$$

$$T_n = \frac{H''(z_n)}{2f'(z_n)} \sim \Big(c_2 - (-1)^{m-1} c_{m+1} \prod_{j=0}^{m-2} e_{t,j} + (3c_3 - 2c_2^2)e_n\Big), \tag{12}$$

and

$$T_n - c_2 \sim \Big(- (-1)^{m-1} c_{m+1} \prod_{j=0}^{m-2} e_{t,j} + (3c_3 - 2c_2^2)e_n\Big). \tag{13}$$

Proof We can calculate the expression of the Hermite interpolation by this way

$$f(z) - H_m(z) = \frac{f^{(m+1)}(\xi)}{(m+1)!}(z - z_n)^2 \prod_{j=0}^{m-2}(z - t_j), (\xi \in I). \tag{14}$$

Now, we get the below mentioned equation by differentiating (14) two times at the point $z = z_n$,

$$H_m''(z_n) = f''(z_n) - 2\frac{f^{(m+1)}(\xi)}{(m+1)!}\prod_{j=0}^{m-2}(z_n - t_j), (\xi \in I). \tag{15}$$

Next the Taylor's series expansion of f' at the point $z_n \in I$ and $\xi \in I$ about the zero α of f provides

$$f'(z_n) = f'(\alpha)\big(1 + 2c_2e_n + 3c_3e_n^2 + O(e_n^3)\big), \tag{16}$$

$$f''(z_n) = f'(\alpha)\big(2c_2 + 6c_3e_n + O(e_n^2)\big), \tag{17}$$

and

$$f^{(m+1)}(\xi) = f'(\alpha)\big((m+1)!c_{m+1} + (m+2)!c_{m+2}e_\xi + O(e_\xi^2)\big), \tag{18}$$

where $e_\xi = \xi - \alpha$. Putting (17), (18) in (15), we obtain

$$H_m''(z_n) = 2f'(\alpha)\left(c_2 - (-1)^{m-1}c_{m+1}\prod_{j=0}^{m-2}e_{t,j} + 3c_3e_n\right), \tag{19}$$

which implies

$$\frac{H_m''(z_n)}{2f'(z_n)} \sim \left(c_2 - (-1)^{m-1}c_{m+1}\prod_{j=0}^{m-2}e_{t,j} + (3c_3 - 2c_2^2)e_n\right). \tag{20}$$

And hence

$$T_n \sim \left(c_2 - (-1)^{m-1}c_{m+1}\prod_{j=0}^{m-2}e_{t,j} + (3c_3 - 2c_2^2)e_n\right), \tag{21}$$

or

$$T_n - c_2 \sim \left(-(-1)^{m-1}c_{m+1}\prod_{j=0}^{m-2}e_{t,j} + (3c_3 - 2c_2^2)e_n\right). \tag{22}$$

The definition of R-order of convergence [21] and the following statement [14,page 287] can be used to estimate the order of convergence of iterative scheme (7). □

Theorem 2 *If the errors $e_j = z_j - \alpha$ evaluated by an iterative root finding method IM fulfill*

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

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

Presently, for the new iterative scheme with memory (7) we can state subsequent convergence theorem:

Theorem 3 *In the iterative method* (7)*, let T_n be the varying parameter and it is calculated by* (8)–(10)*. If an initial guess z_0 is enough near to a simple zero α of $f(x)$, then the R-order of convergence of iterative methods* (8)–(7)*,* (9)–(7) *and* (10)–(7) *with memory is at least* $(6+\sqrt{32})/2 \approx 5.8284$*, 6.0275 and 6.2128, respectively.*

Proof Let the iterative method (IM) generating the sequence of $\{z_n\}$ which is converges to the root α of $f(z)$, by means of R-order $O_R(IM, \alpha) \geq r$, we express

$$e_{n+1} \sim D_{n,r} e_n^r. \tag{23}$$

Next $D_{n,r}$ will tends to the asymptotic error constant D_r of IM by taking $n \to \infty$, then. Therefore

$$e_{n+1} \sim D_{n,r}(D_{n-1,r}e_{n-1}^r)^r = D_{n,r}D_{n-1,r}^r e_{n-1}^{r^2}. \tag{24}$$

The resulting error expression of the with memory scheme (7) can be obtained using (5), (6), and the varying parameter T_n

$$e_{n,w} = w_n - \alpha \sim (c_2 - T_n)e_n^2, \tag{25}$$

and

$$e_{n+1} = z_{n+1} - \alpha \sim B_{n,5}(T_n - c_2)^2 e_n^5 + B_{n,6}(T_n - c_2)e_n^6, \tag{26}$$

where $B_{n,5}$ and $B_{n,6}$ originates from (6) represents the changing quantity which occurs due to the self-accelerating parameter T_n. Also the higher order terms in (25)–(26) are excluded.

Method 1. T_n is evaluated by (8): The calculation of T_n is analogous to the derivation of (23). The R-order convergence of iterative sequence $\{w_n\}$ is take up as p, so

$$e_{n,w} \sim D_{n,p}e_n^p \sim D_{n,p}(D_{n-1,r}e_{n-1}^r)^p = D_{n,p}D_{n-1,r}^p e_{n-1}^{rp}. \tag{27}$$

Using Theorem 2.1 for $m = 2$ and $t_0 = w_{n-1}$, we attain

$$T_n - c_2 \sim c_3 e_{t,0} = c_3 e_{n-1,w}. \tag{28}$$

Now from (25), (26) and (28), we get

$$e_{n,w} \sim -c_3 e_{n-1,w}(D_{n-1,r}e_{n-1}^r)^2 \sim -c_3 D_{n-1,p} D_{n-1,r}^2 e_{n-1}^{2r+p}, \tag{29}$$

and

$$\begin{aligned} e_{n+1} &\sim B_{n,5}c_3^2 e_{n-1,w}^2 e_n^5 + B_{n,6}c_3 e_{n-1,w} e_n^6 \\ &\sim B_{n,5}c_3^2(D_{n-1,p}e_{n-1}^p)^2(D_{n-1,r}e_{n-1}^r)^5 + B_{n,6}c_3(D_{n-1,p}e_{n-1}^p)(D_{n-1,r}e_{n-1}^r)^6, \\ &\sim B_{n,5}c_3^2 D_{n-1,p}^2 D_{n-1,r}^5 e_{n-1}^{5r+2p} + B_{n,6}c_3 D_{n-1,p} D_{n-1,r}^6 e_{n-1}^{6r+p}, \\ &\sim (B_{n,5}c_3^2 D_{n-1,p}^2 D_{n-1,r}^5 + B_{n,6}c_3 D_{n-1,p} D_{n-1,r}^6 e_{n-1}^{r-p}) e_{n-1}^{5r+2p}, \end{aligned} \tag{30}$$

since $r > p$. By equating the exponents of e_{n-1} present in the set of relation (27)–(29) and (24)–(30), we attain the resulting system of equations:

$$\begin{aligned} 2r + p &= rp, \\ 5r + 2p &= r^2. \end{aligned} \tag{31}$$

The solution of system of equation (31) is specified by $r = (6 + \sqrt{32})/2$ and $p = 2.4142$. As a result, the R-order of convergence of the with memory iterative schemes (8)–(7) is at least $(6 + \sqrt{32})/2 \approx 5.8284$.

Method 2. T_n is evaluated by (9): By means of the Theorem 2.1 for $m = 3$, $t_0 = w_{n-1}$ and $t_1 = z_{n-1}$, we get hold of

$$T_n - c_2 \sim -c_4 e_{t,0} e_{t,1} = -c_4 e_{n-1,w} e_{n-1}. \tag{32}$$

In accordance with (25), (26) and (32), we find

$$\begin{aligned} e_{n,w} &\sim (c_2 - T_n) e_n^2 \sim c_4 e_{n-1} e_{n-1,w} (D_{n-1,r} e_{n-1}^r)^2 \\ &\sim c_4 D_{n-1,p} D_{n-1,r}^2 e_{n-1}^{2r+p+1}, \end{aligned} \tag{33}$$

and

$$\begin{aligned} e_{n+1} &\sim B_{n,5} c_4^2 e_{n-1}^2 e_{n-1,w}^2 e_n^5 - B_{n,6} c_4 e_{n-1} e_{n-1,w} e_n^6 \\ &\sim B_{n,5} c_4^2 e_{n-1}^2 (D_{n-1,p} e_{n-1}^p)^2 (D_{n-1,r} e_{n-1}^r)^5 - B_{n,6} c_4 e_{n-1} \\ &\quad (D_{n-1,p} e_{n-1}^p)(D_{n-1,r} e_{n-1}^r)^6 \\ &\sim B_{n,5} c_4^2 D_{n-1,p}^2 D_{n-1,r}^5 e_{n-1}^{5r+2p+2} + B_{n,6}(-c_4) D_{n-1,p} D_{n-1,r}^6 e_{n-1}^{6r+p+1} \\ &\sim e_{n-1}^{5r+2p+1} (B_{n,5} c_4^2 D_{n-1,p}^2 D_{n-1,r}^5 e_{n-1}^1 + B_{n,6}(-c_4) D_{n-1,p} D_{n-1,r}^6 e_{n-1}^{r-p}). \end{aligned} \tag{34}$$

By equating the exponents of e_{n-1} present in the set of relation (27)–(33) and (24)–(34), we attain the resulting system of equations:

$$\begin{aligned} 2r + p + 1 &= rp, \\ 5r + 2p + 1 &= r^2. \end{aligned} \tag{35}$$

The solution of system of equations (35) is specified by $r = 6.0275$ and $p = 2.5967$. As a result, the R-order of convergence of the with memory iterative schemes (9)–(7) is at least 6.0275.

Method 3. T_n is evaluated by (10): By means of the Theorem 2.1 for $m = 4$, $t_0 = w_{n-1}$ and $t_1 = t_2 = z_{n-1}$, we get hold of

$$T_n - c_2 \sim c_5 e_{t,0} e_{t,1} e_{t,2} = c_5 e_{n-1,w} e_{n-1}^2. \tag{36}$$

In accordance with (25), (26) and (36), we find

$$\begin{aligned} e_{n,w} &\sim (c_2 - T_n) e_n^2 \sim -c_5 e_{n-1}^2 e_{n-1,w} (D_{n-1,r} e_{n-1}^r)^2 \\ &\sim -c_5 D_{n-1,p} D_{n-1,r}^2 e_{n-1}^{2r+p+2}, \end{aligned} \tag{37}$$

and

$$\begin{aligned} e_{n+1} &\sim B_{n,5} c_5^2 e_{n-1}^4 e_{n-1,w}^2 e_n^5 + B_{n,6} c_5 e_{n-1}^2 e_{n-1,w} e_n^6 \\ &\sim B_{n,5} c_5^2 e_{n-1}^4 (D_{n-1,p} e_{n-1}^p)^2 (D_{n-1,r} e_{n-1}^r)^5 + B_{n,6} c_5 e_{n-1}^2 \\ &\quad (D_{n-1,p} e_{n-1}^p)(D_{n-1,r} e_{n-1}^r)^6 \\ &\sim B_{n,5} c_5^2 D_{n-1,p}^2 D_{n-1,r}^5 e_{n-1}^{5r+2p+4} + B_{n,6} c_5 D_{n-1,p} D_{n-1,r}^6 e_{n-1}^{6r+p+2} \\ &\sim e_{n-1}^{5r+2p+2} (B_{n,5} c_5^2 D_{n-1,p}^2 D_{n-1,r}^5 e_{n-1}^2 + B_{n,6} c_5 D_{n-1,p} D_{n-1,r}^6 e_{n-1}^{r-p}). \end{aligned} \tag{38}$$

By equating the exponents of e_{n-1} present in the set of relation (27)–(37) and (24)–(38), we attain the resulting system of equations:

$$\begin{aligned} 2r + p + 2 &= rp, \\ 5r + 2p + 2 &= r^2. \end{aligned} \tag{39}$$

The solution of system of equations (39) is specified by $r = 6.2128$ and $p = 2.7673$. As a result, the R-order of convergence of the with memory iterative schemes (10)- (7) is at least 6.2128. □

Numerical Assessment

Within the ensuing segment, after the review of two-step iterative schemes, the comparison has been done between the existing techniques and proposed scheme (7) denoted by MWM5. Our proposed method is compared with existing well known without memory fifth-order schemes HM, ISHM, LLGS, NR, SHM, YCHAM and similar type of two-step with memory methods XW1, XW2 presented in the fres. [2–7, 22] respectively. We now state the above mentioned fifth-order iterative schemes.

In the article [3] Noor and Noor suggested two-step modified Halley method of fifth-order convergence:

$$\begin{aligned} w_n &= z_n - \frac{f(z_n)}{f'(z_n)}, \\ z_{n+1} &= w_n - \frac{2f(z_n)f(w_n)f'(w_n)}{2f(z_n)f'(w_n)^2 - f'(z_n)^2 f(w_n) + f'(z_n)f'(w_n)f(w_n)}, \end{aligned} \tag{40}$$

which is denoted by HM and this is the method which we have modified and converted into with memory method in this paper.

In 2007 Noor and Noor [6] proposed two-step Halley method of fifth-order convergence:

$$\begin{aligned} w_n &= z_n - \frac{2f(z_n)f'(z_n)}{2f'(z_n) - f(z_n)f''(z_n)}, \\ z_{n+1} &= z_n - \frac{2[f(z_n) + f(w_n)]f'(z_n)}{2f'(z_n)^2 - [f(z_n) + f(w_n)]f''(z_n)}, \end{aligned} \tag{41}$$

which is denoted by NR.

In his paper [4, 5], Kou et al. have suggested following iterative method:

$$\begin{aligned} w_n &= z_n - \frac{f(z_n)}{f'(z_n)} - \frac{f''(z_n)f(z_n)^2}{2f'(z_n)^3 - 2f(z_n)f'(z_n)f''(z_n)}, \\ z_{n+1} &= w_n - \frac{f(w_n)}{f'(z_n)} - \frac{f''(z_n)f(w_n)}{2f'(z_n)^3}, \end{aligned} \tag{42}$$

which is denoted by ISHM and

$$\begin{aligned} w_n &= z_n - \frac{f(z_n)}{f'(z_n)} - \frac{f''(z_n)f(z_n)^2}{2f'(z_n)^3 - 2f(z_n)f'(z_n)f''(z_n)}, \\ z_{n+1} &= w_n - \frac{f(w_n)}{f'(z_n) + f''(z_n)(w_n - z_n)}, \end{aligned} \tag{43}$$

which is denoted by SHM.

In 2008 Fang et al. [2] have proposed following fifth-order iterative methods:

$$\begin{aligned} w_n &= z_n - \frac{f(z_n)}{f'(z_n)}, \\ z_{n+1} &= w_n - \frac{5f'(z_n)^2 + 3f'(w_n)^2}{f'(z_n)^2 + 7f'(w_n)^2} \frac{f(w_n)}{f'(z_n)}, \end{aligned} \tag{44}$$

which is denoted by LLGS.

In 2007 Ham and Chun [7] have proposed following fifth-order iterative methods:

$$\begin{aligned} w_n &= z_n - \frac{f(z_n)}{f'(z_n)}, \\ z_{n+1} &= w_n - \frac{f'(w_n) + 3f'(z_n)}{5f'(w_n) - f'(z_n)} \frac{f(w_n)}{f'(z_n)}, \end{aligned} \tag{45}$$

which is denoted by YCHAM.

In 2014, Wang and Zang [22] proposed two with memory methods. Both are two-step methods in the subsequent form:

$$\begin{aligned} w_n &= z_n - \frac{f(z_n)}{T_n f(z_n) + f'(z_n)}, \\ z_{n+1} &= w_n - \left(1 - \frac{f(w_n)}{2T_n f(z_n) + f'(z_n)}\right) \\ &\quad \left(1 + \frac{2f(w_n)}{f(z_n)} + a\left(\frac{2f(w_n)}{f(z_n)}\right)^2\right), \end{aligned} \tag{46}$$

where $a \in R$, which is denoted by XW41, and

$$\begin{aligned} w_n &= z_n - \frac{f(z_n)}{T_n f(z_n) + f'(z_n)}, \\ z_{n+1} &= w_n - \left(1 - \frac{f(w_n)}{2T_n f(z_n) + f'(z_n)}\right) \left(\frac{f(z_n) + (2+b)f(w_n)}{f(z_n) + bf(w_n)}\right), \end{aligned} \tag{47}$$

where $b \in R$, which is denoted by XW42. We are captivating the values of the self-accelerating parameter T_n for both methods in the following form:
Method 4:

$$T_n = -\frac{H_2''(z_n)}{2f'(z_n)}, \tag{48}$$

where $H_2(z) = f(z_n) + f[z_n, z_n](z - z_n) + f[z_n, z_n, w_{n-1}](z - z_n)^2$ and $H_2''(z) = 2f[z_n, z_n, w_{n-1}]$.
Method 5:

$$T_n = -\frac{H_3''(z_n)}{2f'(z_n)}, \tag{49}$$

where $H_3(z) = H_2(z) + f[z_n, z_n, w_{n-1}, z_{n-1}](z - z_n)^2(z - w_{n-1})$ and $H_3''(z) = 2f[z_n, z_n, w_{n-1}] + 2f[z_n, z_n, w_{n-1}, z_{n-1}](z_n - w_{n-1})$.
Method 6:

$$T_n = -\frac{H_4''(z_n)}{2f'(z_n)}, \tag{50}$$

Table 1 Test functions

Nonlinear function	Root
$f_1 = xe^{x^2} - (sinx)^2 + 3cosx + 5$	$-1.2076\ldots$
$f_2 = x^3 - x^2 - 1$	$1.4655\ldots$
$f_3 = x^2 - e^x - 3x + 2$	$0.2575\ldots$
$f_4 = x^3 - 10$	$2.1544\ldots$
$f_5 = cosx - x$	$0.7390\ldots$
$f_6 = sinx - \frac{x}{2}$	$1.8954\ldots$

where $H_4(z) = H_3(z) + f[z_n, z_n, w_{n-1}, z_{n-1}, z_{n-1}](z - z_n)^2(z - w_{n-1})(z - z_{n-1}))$ and $H_4''(z) = 2f[z_n, z_n, w_{n-1}] + (4f[z_n, z_n, w_{n-1}, z_{n-1}] - 2f[z_n, w_{n-1}z_{n-1}, z_{n-1}])(z_n - w_{n-1})$.

In Table 1, we have considered six different nonlinear functions with their zeroes (α). Also in the same table we have mentioned the roots each function only up to four decimal places instead of the infinite number of digits after the decimal (these examples have been taken from the references [3, 15, 23]). Tables 2 and 3 contains the absolute errors $|z_k - \alpha|$ in the first five iteration and the computational order of convergence for the new scheme and existing schemes. The formula of computational order of convergence (COC) given by [16]

$$COC \approx \frac{ln|f(z_{n+1})/f(z_n)|}{ln|f(z_n)/f(z_{n-1})|},$$

is considered to approximate the computational efficiency which verified all the theoretical rate of convergence. In Tables 2 and 3, we have written the numerical results proposed two-step with memory iterative methods which are in accordance with the theoretical part of the paper. To compute the exact root upto 10000 significant digits we have used *"Set Accuracy"* command in Mathematica 8. The nonlinear functions have been solved by the proposed iterative method MWM5 (7) and the computed results are compared with other existing two-step similar type with memory and fifth order without memory iterative schemes.

From Tables 2 and 3 it is clear that the $R - order$ convergence of the proposed with memory scheme MWM5 (7) is accelerated from 5 to 6.2128, in accordance with the used self-accelerating parameter methods given by (8)–(10). The acceleration of convergence order of the proposed scheme is achieved without any extra function evaluation, that can be easily seen from the existing without memory fifth order scheme presented in [3] and new method (7) along with the applied accelerating method (8)–(10). In the table "NC" stands for non-convergence of the methods. The results attained in Tables 2 and 3 by means of the proposed scheme verify the theoretical analysis and also we can notice that our scheme is quite superior to the other iterative schemes.

Conclusion

In this article for calculating the root of nonlinear equations, we have developed the novel two-step iterative scheme with memory. Since the main objective of the paper is to develop the scheme of higher order convergence without any extra calculation. That is done using three different estimations of self-accelerating parameters, without any extra function calculation in the fifth-order scheme to attain higher-order convergence. These parameters are calculated using Hermite interpolation. The acceleration in the R-order of convergence of novel with

Table 2 Comparison of absolute errors

Fun.	Method	Guess	$\|z_1 - \alpha\|$	$\|z_2 - \alpha\|$	$\|z_3 - \alpha\|$	$\|z_4 - \alpha\|$	$\|z_5 - \alpha\|$	COC
f_1	$MWM5$ (7)–(8), $T_0 = 0.03$	−0.8	8.2188 e−2	9.5958e−6	2.5585e−29	1.1746e−166	4.1142e−967	5.8284
	$MWM5$ (7)–(9), $T_0 = 0.03$		8.2188 e−2	8.6989e−7	1.6405e−37	8.6189e−227	3.4185e−1393	6.1623
	$MWM5$ (7)–(10), $T_0 = 0.03$		8.2188 e−2	4.1349e−8	1.3262e−47	1.3262e−303	2.6752e−1957	6.4630
	$XW41$ (46)–(48), $T_0 = 0.03$		NC					
	$XW41$ (46)–(49), $T_0 = 0.03$		NC					
	$XW41$ (46)–(50), $T_0 = 0.03$		NC					
	$XW42$ (47)–(48), $T_0 = 0.03$		7.6738 e−3	1.4479e−8	7.1052e−36	1.7299e−160	6.1445e−729	4.5617
	$XW42$ (47)–(49), $T_0 = 0.03$		7.6738 e−3	2.3432e−9	6.6780e−39	1.5977e−167	1.6402e−728	4.3616
	$XW42$ (47)–(50), $T_0 = 0.03$		7.6738 e−3	4.9207e−10	7.3412e−42	5.5453e−180	2.6115e−782	4.3608
	HM		7.4326 e−2	1.4412e−5	4.2129e−24	8.9940e−117	3.9884e−580	5.0000
	$ISHM$		2.0819	1.7149e−2	1.7608e−9	1.9606e−30	2.7067e−93	3.0000
	$LLGS$		2.5092 e−1	1.8112e−2	2.6736e−8	1.8170e−37	2.6344e−183	5.0000
	NR		NC					
	SHM		4.0781 e−2	1.1158e−6	2.0803e−29	4.6856e−143	2.7163e−711	5.0000
	$YCHAM$		8.2013 e−2	1.1632e−5	6.6421e−25	4.0322e−151	3.3245e−602	5.0000
f_2	$MWM5$ (7)–(8), $T_0 = 0.07$	1.0	7.1243 e−2	8.8403e−9	5.1612e−49	1.7331e−283	4.2561e−1605	5.8284
	$MWM5$ (7)–(9), $T_0 = 0.07$		7.1243 e−2	4.1246e−9	1.1704e−59	1.17339e−413	2.7156e−2890	7.0000
	$MWM5$ (7)–(10), $T_0 = 0.07$		7.1243 e−2	4.1246e−9	4.1704e−59	1.17339e−413	2.7156e−2890	7.0000
	$XW41$ (46)–(48), $T_0 = 0.07$		NC					
	$XW41$ (46)–(49), $T_0 = 0.07$		NC					
	$XW41$ (46)–(50), $T_0 = 0.07$		NC					
	$XW42$ (47)–(48), $T_0 = 0.07$		9.8674 e−2	3.0921e−6	4.8801e−27	8.2580e−122	4.5908e−554	4.5610
	$XW42$ (47)–(49), $T_0 = 0.07$		9.8674 e−2	6.8890e−7	1.4112e−29	4.8457e−128	1.8644e−557	4.3611

Table 2 continued

Fun.	Method	Guess	$\lvert z_1-\alpha\rvert$	$\lvert z_2-\alpha\rvert$	$\lvert z_3-\alpha\rvert$	$\lvert z_4-\alpha\rvert$	$\lvert z_5-\alpha\rvert$	COC
	$XW42$ (47)–(50), $T_0=0.07$		9.8674 e−2	6.8890e−7	2.8189e−29	3.0890e−126	4.9249e−549	4.3605
	HM		8.5831e−2	1.1986e−6	9.8770e−31	3.7538e−151	2.9762e−753	5.0000
	$ISHM$		NC					
	$LLGS$		NC					
	NR		NC					
	SHM		2.3218e−2	1.7531e−9	4.0267e−45	2.5740e−223	2.7472e−1114	5.0000
	$YCHAM$		3.1173e−1	1.3762e−2	5.4550e−10	5.0501e−47	3.4340e−232	5.0000
f_3	$MWM5$ (7)–(8), $T_0=0.03$	-3.3	2.5525 e−3	3.4700e−17	2.1087e−99	8.2513e−579	4.8731e−3372	5.8285
	$MWM5$ (7)–(9), $T_0=0.03$		2.5525 e−3	9.1861e−18	5.7868e−97	1.4468e−672	5.6557e−4144	6.1624
	$MWM5$ (7)–(10), $T_0=0.03$		2.5525 e−3	2.0703e−18	3.0788e−100	9.6661e−773	7.6808e−4995	6.4653
	$XW41$ (46)–(48), $T_0=0.03$		1.2460e−1	7.7.5101e−7	6.5448e−31	1.2151e−140	3.5290e−641	4.5615
	$XW41$ (46)–(49), $T_0=0.03$		1.2460e−1	3.5483e−7	3.0849e−31	4.2917e−136	1.0041e−593	4.3643
	$XW41$ (46)–(50), $T_0=0.03$		1.2460e−1	1.8819e−7	2.8224e−32	3.6916e−140	6.5491e−611	4.3635
	$XW42$ (47)–(48), $T_0=0.03$		1.5396 e−1	9.5958e−6	2.5585e−29	1.1746e−134	4.1142e−912	4.5615
	$XW42$ (47)–(49), $T_0=0.03$		1.5396 e−1	7.8100e−7	8.7539e−30	9.0758e−130	4.0911e−566	4.3641
	$XW42$ (47)–(50), $T_0=0.03$		1.5396 e−1	4.3867e−7	1.0436e−30	2.4695e−133	4.0450e−581	4.3633
	HM		2.7227e−2	1.0659e−11	1.0284e−58	8.5988e−294	3.5144e−1496	5.0000
	$ISHM$		2.3378e−1	6.9291e−7	3.2883e−23	3.5144e−72	4.2903e−219	3.0000
	$LLGS$		1.4633e−1	1.4835e−8	1.6629e−43	2.9420e−218	5.9400e−1092	5.0000
	NR		NC					
	SHM		3.4589e−2	5.0787e−10	3.3007e−49	3.8270e−245	8.0188e−1225	5.0000
	$YCHAM$		6.9815e−2	6.9835e−10	6.3775e−50	4.0850e−250	4.1877e−1251	5.0000

Table 3 Numerical comparison

Fun.	Method	Guess	$\|x_1 - \alpha\|$	$\|x_2 - \alpha\|$	$\|x_3 - \alpha\|$	$\|x_4 - \alpha\|$	$\|x_5 - \alpha\|$	COC
f_4	$MWM5$ (7)–(8), $T_0 = 0.06$	1.1	1.8394 e−1	5.9267e−8	1.1458e−45	1.9691e−265	2.6234e−1546	5.8284
	$MWM5$ (7)–(9), $T_0 = 0.06$		1.8394 e−1	3.5520e−8	4.7556e−55	4.7556e−382	5.9574e−2680	7.0000
	$MWM5$ (7)–(10), $T_0 = 0.06$		1.8394 e−1	3.5520e−8	4.7556e−55	4.7556e−382	5.9574e−2680	7.0000
	$XW41$ (46)–(48), $T_0 = 0.06$		NC					
	$XW41$ (46)–(49), $T_0 = 0.06$		NC					
	$XW41$ (46)–(50), $T_0 = 0.06$		NC					
	$XW42$ (47)–(48), $T_0 = 0.06$		1.9607 e−1	9.5958e−5	2.5585e−24	1.1746e−111	4.1142e−508	4.5611
	$XW42$ (47)–(49), $T_0 = 0.06$		1.9607 e−1	1.4103e−6	6.6321e−29	6.5007e−127	4.0309e−549	4.3631
	$XW42$ (47)–(50), $T_0 = 0.06$		1.9607 e−1	1.4103e−6	1.3209e−28	4.0938e−124	1.0106e−540	4.3620
	HM		2.6605e−1	1.6676e−5	2.9932e−26	5.5757e−130	1.2507e−648	5.0000
	$ISHM$		NC					
	$LLGS$		NC					
	NR		NC					
	SHM		9.5579e−2	1.3895e−7	8.0132e−37	5.1119e−183	5.4009e−914	5.0000
	$YCHAM$		1.0340	8.6613e−1	1.9934e−1	2.4996e−5	5.2845e−25	5.0900
f_5	$MWM5$ (7)–(8), $T_0 = 0.05$	-0.4	6.1447 e−2	1.9947e−10	4.0233e−60	9.5666e−350	8.5738e−2028	5.8284
	$MWM5$ (7)–(9), $T_0 = 0.05$		6.1447 e−2	5.5202e−12	6.4747e−73	1.3770e−448	1.4948e−2763	6.1622
	$MWM5$ (7)–(10), $T_0 = 0.05$		6.1447 e−2	3.5446e−12	5.8868e−78	2.0852e−503	1.8864e−3253	6.4683
	$XW41$ (46)–(48), $T_0 = 0.05$		NC					
	$XW41$ (46)–(49), $T_0 = 0.05$		NC					
	$XW41$ (46)–(50), $T_0 = 0.05$		NC					
	$XW42$ (47)–(48), $T_0 = 0.05$		1.9816 e−1	1.8754e−5	2.4242e−24	1.5924e−110	1.1689e−503	4.5616
	$XW42$ (47)–(49), $T_0 = 0.05$		1.9816 e−1	5.8142e−6	2.1132e−26	3.0314e−115	1.0580e−502	4.3611
	$XW42$ (47)–(50), $T_0 = 0.05$		1.9816 e−1	1.9336e−6	1.9253e−28	1.0289e−123	4.1908e−539	4.3600

Table 3 continued

Fun.	Method	Guess	$\lvert x_1 - \alpha \rvert$	$\lvert x_2 - \alpha \rvert$	$\lvert x_3 - \alpha \rvert$	$\lvert x_4 - \alpha \rvert$	$\lvert x_5 - \alpha \rvert$	COC
	HM		1.4737e−1	4.2953e−7	7.1725e−35	9.3128e−174	3.4365e−865	5.0000
	$ISHM$		5.8930e−1	8.1221e−4	1.6942e−12	1.5365e−38	1.1462e−116	3.0000
	$LLGS$		5.4378e−1	2.6355e−3	1.4853e−15	8.3781e−77	4.7842e−383	5.0000
	NR		NC					
	SHM		1.0569e−2	1.8137e−12	2.6495e−61	1.7626e−127	2.2986e−1526	5.0000
	$YCHAM$		2.3893e−1	2.8391e−5	1.3762e−25	3.3748e−127	2.9928e−635	5.0000
f_6	$MWM5$ (7)–(8), $T_0 = 0.04$	-0.8	5.6080 e−1	1.5570e−3	2.9871e−18	5.0692e−104	6.3120e−604	5.8284
	$MWM5$ (7)–(9), $T_0 = 0.04$		5.6080 e−1	1.2190e−3	1.2366e−21	1.7990e−138	1.7805e−893	6.4620
	$MWM5$ (7)–(10), $T_0 = 0.04$		5.6080 e−1	1.2802e−3	4.4285e−24	4.3962e−153	1.7416e−988	6.4758
	$XW41$ (46)–(48), $T_0 = 0.04$		NC					
	$XW41$ (46)–(49), $T_0 = 0.04$		NC					
	$XW41$ (46)–(50), $T_0 = 0.04$		NC					
	$XW42$ (47)–(48), $T_0 = 0.04$		2.1910	1.8954	1.8954	1.8954	1.8954	4.5637
	$XW42$ (47)–(49), $T_0 = 0.04$		2.1910	1.8954	1.8954	1.8954	1.8954	4.3561
	$XW42$ (47)–(50), $T_0 = 0.04$		2.1910	1.8954	1.8954	1.8954	1.8954	4.3552
	HM		NC					
	$ISHM$		1.4978	6.0437e−2	1.1760e−5	1.1120e−16	9.4015e−50	3.0000
	$LLGS$		NC					
	NR		NC					
	SHM		5.2170e−2	1.2770e−7	1.1320e−35	6.1974e−176	3.0473e−877	5.0000
	$YCHAM$		NC					

memory iterative schemes are 5 to 5.8284, 6.0275 and 6.2128. The numerical assessment of the new scheme by means of the different test examples verifies that it supports the main theorem.

Acknowledgements The authors would like to pay their sincere gratitude to the reviewers for their useful suggestions.

Funding The authors have not disclosed any funding.

Data Availability Enquiries about data availability should be directed to the authors.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

1. Wang, X., Zhang, T.: Higher-order Newton-type iterative methods with and without memory for solving nonlinear equations. Math. Commun. **19**(1), 91–109 (2014)
2. Fang, L., Sun, L., He, G.: An efficient Newton type method with fifth order convergence for solving nonlinear equation. Comput. Appl. Math. **27**(3), 269–274 (2008)
3. Noor, M.A., Khan, W.A., Hussain, A.: A new modified Halley method without second derivatives for nonlinear equation. Appl. Math. Comput. **189**, 1268–1273 (2007)
4. Kou, J., Li, Y.: Improvements of Chebyshev–Halley methods with fifth-order convergence. Appl. Math. Comput. **188**(1), 143–147 (2007)
5. Kou, J., Li, Y., Wang, X.: A family of fifth-order iterations composed of Newton and third-order methods. Appl. Math. Comput. **186**(2), 1258–1262 (2007)
6. Noor, M.A., Noor, K.I.: Fifth-order iterative methods for solving nonlinear equations. Appl. Math. Comput. **188**(1), 406–410 (2007)
7. Ham, Y., Chun, C.: A fifth-order iterative method for solving nonlinear equations. Appl. Math. Comput. **194**(1), 287–290 (2007)
8. Soleymani, F.: Some optimal iterative methods and their with memory variants. J. Egypt. Math. Soc. **21**(2), 133–141 (2013)
9. Choubey, N., Cordero, A., Jaiswal, J.P., Torregrosa, J.R.: Dynamical techniques for analyzing iterative schemes with memory. Complexity **2018**, 1–13 (2018)
10. Choubey, N., Jaiswal, J.P.: Two and three point with memory methods for solving nonlinear equations. Numer. Anal. Appl. **10**(1), 71–89 (2017)
11. Choubey, N., Pandey, B., Jaiswal, J.P.: Several two-point with memory iterative methods for solving nonlinear equations. Afr. Mat. **29**, 435–449 (2018)
12. Kumar, S., Kanwar, V., Tomar, S.K., Singh, S.: Geometrically constructed families of Newton's method for unconstrained optimization and nonlinear equations. Int. J. Math. Math. Sci. **2011**, 1–9 (2011)
13. Behl, R., Kanwar, V.: New highly efficient families of higher-order methods for simple roots, permitting $f'(x_n) = 0$. Int. J. Math. Math. Sci. **2014**, 1–12 (2014)
14. Alefeld, G., Herzberger, J.: Introduction to Interval Computation. Academic Press, New York (1983)
15. Chun, C., Lee, M.Y.: A new optimal eighth-order family of iterative methods for the solution of nonlinear equations. Appl. Math. Comput. **223**, 506–519 (2013)
16. Weerakoon, S., Fernando, T.G.I.: A variant of newton's method with accelerated third-order convergence. Appl. Math. Lett. **13**(8), 87–93 (2000)
17. Soheili, A.R., Ahmadian, S.A., Naghipoor, J.: A family of predictor-corrector methods based on weight combination of quadratures for solving nonlinear equations. Int. J. Nonlinear Sci. **6**, 29–33 (2008)
18. Naghipoor, J., Ahmadian, S.A., Soheili, A.R.: An improved regular Falsi method for finding simple zeros of solving nonlinear equations. Appl. Math. Sci. **2**(8), 381–386 (2008)
19. Traub, J.F.: Iterative Methods for Solution of Equations. Prentice Hall, Englewood Cliffs, NJ (1964)
20. Petkovic, M.S., Neta, B., Petkovic, J.D., Dzunice, J.: Multipoint Point Methods for Solving Nonlinear Equations. Academic Press, New York (2013)

21. Ortega, J., Rheinbolt, W.: Iterative Solution of Nonlinear Equations in Several Variables. Academic Press, New York (2000)
22. Wang, X., Zhang, T.: A new family of Newton-type iterative methods with and without memory for solving nonlinear equations. Calcolo **51**(1), 1–15 (2014)
23. Wang, X., Zhang, T.: Some Newton-type iterative methods with and without memory for solving nonlinear equations. Int. J. Comput. Meth. **11**(5), 1–20 (2013)