Journal of Applied
and Numerical Analysis

2025, Vol. 3, pp. 70–87
doi: 10.30970/ana.2025.3.70

Design of new derivative-free iterative methods with memory for solving system of nonlinear equations

Nishant Kumar[1], J. P. Jaiswal[1]

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

Анотація. У цій статті ми представляємо безпохідні методи з пам'яттю, які демонструють зростаючі порядки збіжності для розв'язування систем нелінійних рівнянь. Ці методи з пам'яттю побудовано на основі існуючого методу четвертого порядку без пам'яті шляхом підбирання чотирьох різних вільних параметрів, що самокорегуються, на кожному ітераційному кроці. Порядок збіжності нових методів з пам'яттю покращено з 4 до приблизно 4.23607, 4.44949, 4.56155 та 5.23607. Обчислювальну ефективність запропонованих методів також досліджено порівняно з добре відомими методами подібної природи. Продемонстровано, що нові методи загалом мають вищу ефективність, і цей висновок підтверджено чисельними експериментами.

Abstract. In the present article, we introduce derivative-free methods with memory that exhibit increasing convergence orders for solving systems of nonlinear equations. These with-memory methods are built upon the existing fourth-order method without memory by adjusting four different free self-corrected parameters in each iterative step. The convergence order of the new methods with memory is enhanced from 4 to approximately 4.23607, 4.44949, 4.56155 and 5.23607. The computational efficiency of the proposed techniques is also examined over well existing similar nature methods. It is demonstrated that these new methods generally offer greater efficiency and a conclusion is supported by numerical experiments.

1 Introduction

When tackling real-world problems in engineering or science, it is common to encounter situations where one needs to find the solution of a nonlinear system of equations. This is typically expressed in the form

$$F(x) = 0, \tag{1.1}$$

where $F : \mathbb{R}^m \to \mathbb{R}^m$ and $D \subset \mathbb{R}^m$ is a neighborhood of a solution to the system (1.1). These equations are often challenging to solve exactly, so instead, we aim for approximate solutions. Iterative methods come into play here, generating a sequence of approximations denoted as $x^{(k)}$, which under certain conditions converge to the solution of the system.

Among these methods, Newton's procedure stands out as one of the most prominent. It's expressed as

$$x^{(k+1)} = x^{(k)} - [F'(x^{(k)})]^{-1} F(x^{(k)}), \tag{1.2}$$

denoting as $F'(x^{(k)})$ the Jacobian matrix related to F. Newton's method (1.2) is renowned for its efficiency and straightforwardness along with its remarkable quadratic convergence. Steffensen's

Key words: derivative-free methods, methods with memory, nonlinear systems of equations, self-correcting parameters

method, an alternative approach emerges when the derivative in Newton's scheme is replaced by the divided difference $[x + F(x), x; F]$. Steffensen's method, highlighted in the ref. [17], is derivative-free and also exhibits quadratic convergence. However, a potential drawback of methods like Newton's is the computation of the inverse of Jacobian matrix in the iterative expression. This step can pose challenges when the function under study is non-differentiable, its derivative is computationally expensive or the Jacobian matrix is singular.

Several Newton-like methods have been developed, utilizing diverse techniques such as weight functions, direct composition, and Jacobian matrix approximations via divided difference operators. As a result, various iterative strategies for solving $F(x) = 0$ have been analyzed, each demonstrating different convergence orders. These approaches are designed to accelerate convergence or improve computational efficiency.

In recent studies [5], [7], novel parametric families of iterative techniques have been introduced, presenting a rapid approach for solving such problems. Additionally, earlier contributions [2], [14] presented various iterative methodologies that circumvent the need for the Jacobian matrix, demonstrating favorable orders of convergence. These methods substitute the Jacobian matrix $F'(-)$ with the divided difference operator $[-, -; F]$.

The primary objective of this research endeavor is to devise derivative-free methodologies characterized by both high convergence orders and low computational expenses. To achieve this goal, we introduce novel derivative-free techniques boasting fourth-order convergence. These methods are designed to minimize the utilization of function evaluations and inverse operators in their iterative schemes, thereby ensuring minimal computational burden and maximizing computational efficiency.

Our presentation is structured as follows: In Section 2, we introduce the new methods and conduct a convergence analysis. Section 3 evaluates the computational efficiency of these methods and compares them with some well existing algorithms in general terms. Section 4 offers a variety of numerical examples to validate the theoretical results and compare the convergence properties of the proposed methods with those of similar nature some well existing methods. Finally, Section **??** provides concluding remarks.

2 With memory method and its convergence analysis

In the subsequent section, we aim to refine the parameter β within the iterative method introduced by Sharma and Arora [16] in order to enhance its convergence rate. Initially, we consider the fourth-order method without memory, as outlined in the referenced article [16]

$$\begin{aligned} y^{(k)} &= x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}), \\ x^{(k+1)} &= y^{(k)} - \Big(3I - [w^{(k)}, x^{(k)}; F]^{-1} \Big([y^{(k)}, x^{(k)}; F] + [y^{(k)}, w^{(k)}; F]\Big)\Big) [w^{(k)}, x^{(k)}; F]^{-1} F(y^{(k)}). \end{aligned} \tag{2.1}$$

In this approach, $w^{(k)}$ is defined as $w^{(k)} = x^{(k)} + \beta F(x^{(k)})$, which is represented by M_4. The corresponding error expression for this method is given by

$$e^{(k+1)} = (I + \beta F'(\alpha))[A_2^2(5I + \beta F'(\alpha)(5I + \beta F'(\alpha)) - A_3(I + \beta F'(\alpha))]A_2(e^{(k)})^4 + O((e^{(k)})^5), \tag{2.2}$$

where $e^{(k+1)} = x^{(k+1)} - \alpha$, $A_i = \frac{1}{i!}[F'(\alpha)]^{-1}F^{(i)}(\alpha)$. Let α be a solution of the nonlinear system $F(x) = 0$. It is clear from the error equation (2.2) that the convergence order of the method is 4 when $\beta \neq -[F'(\alpha)]^{-1}$. Now, if we assume $\beta = -[F'(\alpha)]^{-1}$, it can be shown that the method's convergence order increases upto 5. Since the exact value of $F'(\alpha)$ is unknown, we can only use an approximation of $F'(\alpha)$, based on the available data, to enhance the convergence rate to a certain extent.

The core concept in developing memory-based methods involves iteratively determining the parameter matrix $\beta = B_i^{(k)}$ (for $k = 1, 2, \ldots$). This is done by using an approximation of $[F'(\alpha)]^{-1}$

that is sufficiently accurate and based on the available data. We propose the following four forms for the varying matrix parameter $B_i^{(k)}$ $(i = 1, 2, 3, 4)$

Scheme 1.

$$B_1^{(k)} = -[w^{(k-1)}, x^{(k-1)}; F]^{-1}.$$

Scheme 2.

$$B_2^{(k)} = -[2x^{(k)} - x^{(k-1)}, x^{(k-1)}; F]^{-1}.$$

This divided difference operator is also known as Kurchatov's divided difference.

Scheme 3.

$$B_3^{(k)} = -[x^{(k)}, y^{(k-1)}; F]^{-1}.$$

Scheme 4.

$$B_4^{(k)} = -[2x^{(k)} - y^{(k-1)}, y^{(k-1)}; F]^{-1}.$$

By replacing the parameter β in method (2.1) with the schemes 1 to 4, we derive four new iterative methods with memory, which are detailed as follows

$$\begin{cases} w^{(k)} = x^{(k)} + B_1^{(k)} F(x^{(k)}), \\ y^{(k)} = x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}), \\ x^{(k+1)} = y^{(k)} - \left(3I - [w^{(k)}, x^{(k)}; F]^{-1} \left([y^{(k)}, x^{(k)}; F] + [y^{(k)}, w^{(k)}; F]\right)\right) [w^{(k)}, x^{(k)}; F]^{-1} F(y^{(k)}), \end{cases} \tag{2.3}$$

$$\begin{cases} w^{(k)} = x^{(k)} + B_2^{(k)} F(x^{(k)}), \\ y^{(k)} = x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}), \\ x^{(k+1)} = y^{(k)} - \left(3I - [w^{(k)}, x^{(k)}; F]^{-1} \left([y^{(k)}, x^{(k)}; F] + [y^{(k)}, w^{(k)}; F]\right)\right) [w^{(k)}, x^{(k)}; F]^{-1} F(y^{(k)}), \end{cases} \tag{2.4}$$

$$\begin{cases} w^{(k)} = x^{(k)} + B_3^{(k)} F(x^{(k)}), \\ y^{(k)} = x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}), \\ x^{(k+1)} = y^{(k)} - \left(3I - [w^{(k)}, x^{(k)}; F]^{-1} \left([y^{(k)}, x^{(k)}; F] + [y^{(k)}, w^{(k)}; F]\right)\right) [w^{(k)}, x^{(k)}; F]^{-1} F(y^{(k)}), \end{cases} \tag{2.5}$$

$$\begin{cases} w^{(k)} = x^{(k)} + B_4^{(k)} F(x^{(k)}), \\ y^{(k)} = x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}), \\ x^{(k+1)} = y^{(k)} - \left(3I - [w^{(k)}, x^{(k)}; F]^{-1} \left([y^{(k)}, x^{(k)}; F] + [y^{(k)}, w^{(k)}; F]\right)\right) [w^{(k)}, x^{(k)}; F]^{-1} F(y^{(k)}). \end{cases} \tag{2.6}$$

To assess the convergence characteristics of the schemes (2.3), (2.4), (2.5), and (2.6), we require the following results concerning the Taylor expansions of vector functions (for a detailed explanation, see ref. [12]).

Lemma 2.1. *Let $F : D \subset \mathbb{R}^m \to \mathbb{R}^m$ be p-times Frechet differentiable in a convex set $D \subset \mathbb{R}^m$. Then for any $x, h \in \mathbb{R}^m$, the following expression holds*

$$F(x + h) = F(x) + F'(x)h + \frac{1}{2!}F''(x)h^2 + \frac{1}{3!}F'''(x)h^3 + \cdots + \frac{1}{p-1!}F^{(p-1)}(x)h^{p-1} + R_p,$$

where

$$\|R_p\| \leq \frac{1}{p!} \sup_{0 \leq t \leq 1} \|F^{(p)}(x + th)\| \|h\|^p \text{ and } h^p = \underbrace{(h, h, \cdots, h)}_{p-times}.$$

In our method, we have utilized the divided difference operator for the multi-variable function F (see [1, 8, 12]). This operator, denoted by $[-, -; F]$, is a mapping $D \times D \subset \mathbb{R}^m \times \mathbb{R}^m \to L(\mathbb{R}^m)$, and its definition can be summarized as follows

$$[x+h,x;F]=\int_0^1 F'(x+th)dt;\ \forall\ x,h\in\mathbb{R}^m.$$

Expanding $F'(x+th)$ in a Taylor series at the point x and integrating, we obtain

$$[x+h,x;F]=\int_0^1 F'(x+th)dt=F'(x)+\frac{1}{2}F''(x)h+O(\|h\|^2). \tag{2.7}$$

Let $e^{(k)}=x^{(k)}-\alpha$ denote the error in approximating the solution α at the kth iteration. Assuming that $[F'(\alpha)]^{-1}$ exists, we expand $F(x^{(k)})$ and its first two derivatives in the vicinity of α. This expansion leads to the following expressions

$$F(x^{(k)})=F'(\alpha)(e^{(k)}+A_2(e^{(k)})^2+A_3(e^{(k)})^3+O(\|e^{(k)}\|^4)).$$

Hence

$$F'(x^{(k)})=F'(\alpha)(I+2A_2(e^{(k)})+3A_3(e^{(k)})^2+O(\|e^{(k)}\|^3)) \tag{2.8}$$

and

$$F''(x^{(k)})=F'(\alpha)(2A_2+6A_3(e^{(k)})+O(\|e^{(k)}\|^2)). \tag{2.9}$$

The subsequent theorems analyze the convergence rate of the iterative schemes described in (2.3)–(2.6):

Theorem 2.2. *Consider the function $F:D\subset\mathbb{R}^m\to\mathbb{R}^m$ is differentiable within a neighborhood D of the root α of F. Suppose that $F'(\alpha)$ is continuous and $[F'(\alpha)]^{-1}$ exists. If an initial guess $x^{(0)}$ is sufficiently close to α, the sequence $\{x^{(k)}\}$ generated by method (2.3) converges to α with a convergence order of* 4.23607. *Similarly, the method (2.4) yields convergence to the root α with a convergence order of* 4.44949.

Proof. Suppose $\{x^{(k)}\}$ is a sequence of approximations generated by an iterative scheme converging to the zero α of F with an R-order of at least r. Then

$$e^{(k+1)}\sim D^{(k,r)}(e^{(k)})^r,$$

$$e^{(k+1)}\sim D^{(k,r)}(D^{(k-1,r)}(e^{(k-1)})^r)^r\sim D^{(k,r)}(D^{(k-1,r)})^r(e^{(k-1)})^{r^2}. \tag{2.10}$$

Let

$$\begin{aligned}e_w^{(k)}&=w^{(k)}-\alpha\\&=x^{(k)}+B^{(k)}F(x^{(k)})-\alpha\\&=e^{(k)}+B^{(k)}F(x^{(k)})\\&=e^{(k)}+B^{(k)}F'(\alpha)[e^{(k)}+O(\|e^{(k)}\|^2)]\\&=(I+B^{(k)}F'(\alpha))e^{(k)}+O(\|e^{(k)}\|^2).\end{aligned} \tag{2.11}$$

The error expression (2.2), may be written as

$$e^{(k+1)}=L^{(k)}(I+B^{(k)}F'(\alpha))(e^{(k)})^4+O(\|e^{(k)}\|^5), \tag{2.12}$$

where $L^{(k)}=[A_2^2(5I+\beta F'(\alpha)(5I+\beta F'(\alpha))-A_3(I+\beta F'(\alpha))]A_2$. By setting $x+h=w^{(k-1)}$ and $x=x^{(k-1)}$ in (2.7), and utilizing (2.8) and (2.9), we derive

$$[w^{(k-1)},x^{(k-1)};F]=F'(\alpha)(I+A_2(e_w^{(k-1)}+e^{(k-1)})+O(\|e^{(k-1)}\|^2)).$$

Now,

$$[w^{(k-1)},x^{(k-1)};F]^{-1}=(I-A_2(e_w^{(k-1)}+e^{(k-1)})+O(\|e^{(k-1)}\|^2))F'(\alpha)^{-1},$$

so

$$B^{(k)} = -(I - A_2(e_w^{(k-1)} + e^{(k-1)}) + O(\|e^{(k-1)}\|^2))F'(\alpha)^{-1}$$

and hence

$$I + B^{(k)}F'(\alpha) = A_2(e_w^{(k-1)} + e^{(k-1)}) + O(\|e^{(k-1)}\|^2). \tag{2.13}$$

From (2.11) for $k-1$ index, we have

$$e_w^{(k-1)} = (I + B^{(k-1)}F'(\alpha))e^{(k-1)} + O(\|e^{(k-1)}\|^2). \tag{2.14}$$

By applying equation (2.14) to equation (2.13), we obtain

$$I + B^{(k)}F'(\alpha) = A_2((I + B^{(k-1)}F'(\alpha))e^{(k-1)} + e^{(k-1)}) + O(\|e^{(k-1)}\|^2)$$

or

$$I + B^{(k)}F'(\alpha) = A_2(2I + B^{(k-1)}F'(\alpha))e^{(k-1)} + O(\|e^{(k-1)}\|^2). \tag{2.15}$$

Substituting relation (2.15) in (2.12), we obtain

$$e^{(k+1)} = L^{(k)}A_2(2I + B^{(k-1)}F'(\alpha))e^{(k-1)}(e^{(k)})^4 + O(\|e^{(k)}\|^5).$$

Therefore, when $x^{(k)} \to \alpha$ and letting $k \to \infty$, we have

$$\begin{aligned} e^{(k+1)} &\sim (e^{(k)})^4 e^{(k-1)} \\ &\sim (D^{(k-1,r)}(e^{(k-1)})^r)^4 e^{(k-1)} \\ &\sim (D^{(k-1,r)})^4 (e^{(k-1)})^{4r+1}. \end{aligned} \tag{2.16}$$

By comparing the exponents of $e^{(k-1)}$ on the right side of equations (2.16) and (2.10), we derive the indicial equation

$$r^2 = 4r + 1 \Rightarrow r^2 - 4r - 1 = 0 \Rightarrow r = 4.23607.$$

The iterative method (2.3) has a convergence order of 4.23607. Furthermore, by utilizing (2.7), we can express

$$[2x^{(k)} - x^{(k-1)}, x^{(k-1)}; F] = F'(\alpha)(I + 2A_2e^{(k)} + A_3(e^{(k-1)})^2 - 2A_3e^{(k)}e^{(k-1)} + 4A_3(e^{(k)})^2 + O_3(\|e^{(k)}, e^{(k-1)}\|)).$$

Then,

$$\begin{aligned} [2x^{(k)} - x^{(k-1)}, x^{(k-1)}; F]^{-1} = {} & (I - 2A_2e^{(k)} - A_3(e^{(k-1)})^2 + 2A_3e^{(k)}e^{(k-1)} + 4(A_2^2 - A_3)(e^{(k)})^2)F'(\alpha)^{-1} \\ & + O_3(\|e^{(k)}, e^{(k-1)}\|). \end{aligned}$$

Here, $O_3(\|e^{(k)}, e^{(k-1)}\|)$ indicates the sum of all terms of third and higher orders in the error vectors $e^{(k)}$ and $e^{(k-1)}$, including mixed terms like $e^{(k)}(e^{(k-1)})^2$, which appear in the local error expansion. Therefore,

$$I + B^{(k)}F'(\alpha) = 2A_2e^{(k)} + A_3(e^{(k-1)})^2 - 2A_3e^{(k)}e^{(k-1)} - 4(A_2^2 - A_3)(e^{(k)})^2.$$

Thus, we note that the terms $e^{(k)}$, $e^{(k)}e^{(k-1)}$, $(e^{(k)})^2$, or $(e^{(k-1)})^2$ may appear in $I + B^{(k)}F'(\alpha)$. It is evident that $e^{(k)}e^{(k-1)}$ and $(e^{(k)})^2$ diminish more quickly to zero compared to $e^{(k)}$. Hence, it is essential to determine whether $e^{(k)}$ or $(e^{(k-1)})^2$ exhibits a faster rate of convergence. Suppose the method has an R-order of at least p. In this case, we can express

$$e^{(k+1)} \sim D^{(k,p)}(e^{(k)})^p,$$

where $D^{(k,p)}$ tends to D_p, the asymptotic error constant, when $k \to \infty$. Then, we have

$$\frac{e^{(k)}}{(e^{(k-1)})^2} \sim \frac{D^{(k-1,p)}(e^{(k-1)})^p}{(e^{(k-1)})^2}.$$

If $p > 2$, it follows that $\frac{D^{(k-1,p)}(e^{(k-1)})^p}{(e^{(k-1)})^2}$ approaches zero as $k \to \infty$. Consequently, for $p > 2$, we have $I + B^{(k)}F'(\alpha) \sim (e^{(k-1)})^2$. Utilizing the error equation (2.12) along with this relationship, we derive

$$e^{(k+1)} = A_2 L^{(k)}(D^{(k-1,r)})^4(e^{(k-1)})^{4r+2}. \tag{2.17}$$

Equalizing the exponents of $e^{(k-1)}$ in equations (2.10) and (2.17), we deduce that

$$r^2 = 4r + 2.$$

The unique positive solution to this equation gives the convergence order of method (2.4), which is $r \approx 4.44949$. □

The two prior methods, both incorporating memory were devised using the variable $x^{(k-1)}$. Now, we're about to delve into the consequences of employing the approximation $y^{(k-1)}$ instead. In essence, we're about to explore the methods (2.5) and (2.6). Continuing our investigation, we aim to establish the convergence order of the memory-based methods (2.5) and (2.6). This analysis will follow a similar approach to the one previously outlined.

Theorem 2.3. *Consider the function $F : D \subset \mathbb{R}^m \to \mathbb{R}^m$ is differentiable within a neighborhood D of the root α of F. Suppose that $F'(\alpha)$ is continuous and $[F'(\alpha)]^{-1}$ exists. Subsequently, given an initial estimate $x^{(0)}$ sufficiently close to α, we obtain:*

- *the sequence of iterates $x^{(k)}$, generated by method (2.5), converges to α with order* $p = 4.56155$.
- *the sequence of iterates $x^{(k)}$, generated by method (2.6), converges to α with order* $p = 5.23607$.

3 Computational efficiency

To evaluate the efficiency of the offered techniques, we will use the efficiency index

$$E = \frac{1}{\log d}\frac{\log p}{C}.$$

Here, p specifies the convergence order, C represents the computing cost per iteration and d denotes the number of significant decimal digits in the approximation $x^{(k)}$. For a system comprising m nonlinear equations in m unknowns, referencing the insights provided in the article [9], the computational cost per iteration can be expressed as

$$C(\mu, m, l) = A(m)\mu + P(m, l). \tag{3.1}$$

Here, $A(m)$ signifies the count of evaluations of scalar functions involved in the computation of F and $[x, y; F]$, while $P(m, l)$ represents the number of products necessary per iteration. The divided difference $[x, y; F]$ of F takes the form of an $m \times m$ matrix with specific elements, as elucidated in [10]

$$[x, y; F]_{ij} = \frac{f_i(x_1, \cdots, x_j, y_{j+1}, \cdots, y_m) - f_i(x_1, \cdots, x_{j-1}, y_j, \cdots, y_m)}{x_j - y_j}; \; i \geq 1, j \leq m. \tag{3.2}$$

It's worth noting that Grau-Sanchez et al. have proposed a more accurate formulation of the divided difference [11]. However, the form (3) remains the most widely used in practical applications, including the present study, where we adopt the same form. To represent the value of $C(\mu, m, l)$ in terms of products, we examine the ratio $\mu > 0$ between products and evaluations of scalar functions as well as the ratio $l \geq 1$ between products and quotients. Moreover, in computing the computational cost per iteration, we rely on the following considerations:

- The computation of F entails the evaluation of m scalar functions $f_1,\ f_2,\ \cdots,\ f_m$.
- The calculation of the divided difference $[x, y; F]$ involves $m(m-1)$ scalar functions, as $F(x)$ and $F(y)$ are computed separately.
- Each divided difference involves m^2 quotients, m products for vector-scalar multiplication and m^2 products for matrix-scalar multiplication.
- Determining an inverse linear operator involves solving a linear system, which demands $\frac{m(m-1)(2m-1)}{6}$ products and $\frac{m(m-1)}{2}$ quotients in the LU decomposition phase, as well as $m(m-1)$ products and m quotients in resolving two triangular linear systems.

This study systematically evaluates the efficiency of four proposed techniques: $M^1_{4.23607}$ (2.3), $M^1_{4.44949}$ (2.4), $M^1_{4.56155}$ (2.5), and $M^1_{5.23607}$ (2.6), in comparison with existing memory-based techniques, namely M^1_3 and M^1_5 from [4], along with $M^1_{4.236}$, $M^2_{4.236}$, $M^3_{4.236}$, $M^4_{4.236}$, and M^2_5 as reported in [18].
M^1_3 is expressed as

$$x^{(k+1)} = x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}),$$

where $w^{(k)} = x^{(k)} + B^{(k)}_2 F(x^{(k)})$.
M^1_5 is defined as

$$\begin{aligned} y^{(k)} &= x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}),\\ x^{(k+1)} &= y^{(k)} - [w^{(k)}, y^{(k)}; F]^{-1} F(y^{(k)}), \end{aligned}$$

where $w^{(k)} = x^{(k)} + B^{(k)}_2 F(x^{(k)})$.
$M^1_{4.236}$ is given as

$$\begin{aligned} y^{(k)} &= x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}),\\ x^{(k+1)} &= y^{(k)} - [w^{(k)}, y^{(k)}; F]^{-1} F(y^{(k)}), \end{aligned}$$

where $w^{(k)} = x^{(k)} + B^{(k)}_5 F(x^{(k)}),\ B^{(k)}_5 = -[2y^{(k-1)} - x^{(k)}, x^{(k)}; F]^{-1}$.
$M^2_{4.236}$ is expressed as

$$\begin{aligned} y^{(k)} &= x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}),\\ x^{(k+1)} &= y^{(k)} - [w^{(k)}, y^{(k)}; F]^{-1} F(y^{(k)}), \end{aligned}$$

where $w^{(k)} = x^{(k)} + B^{(k)}_6 F(x^{(k)}),\ B^{(k)}_6 = -[2y^{(k-1)} - x^{(k)}, y^{(k-1)}; F]^{-1}$.
$M^3_{4.236}$ is given as

$$\begin{aligned} y^{(k)} &= x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}),\\ x^{(k+1)} &= y^{(k)} - [w^{(k)}, y^{(k)}; F]^{-1} F(y^{(k)}), \end{aligned}$$

where $w^{(k)} = x^{(k)} + B^{(k)}_7 F(x^{(k)}),\ B^{(k)}_7 = -[2x^{(k)} - y^{(k-1)}, x^{(k)}; F]^{-1}$.
$M^4_{4.236}$ is given as

$$\begin{aligned} y^{(k)} &= x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}),\\ x^{(k+1)} &= y^{(k)} - [w^{(k)}, y^{(k)}; F]^{-1} F(y^{(k)}), \end{aligned}$$

where $w^{(k)} = x^{(k)} + B^{(k)}_8 F(x^{(k)}),\ B^{(k)}_8 = -[3x^{(k)} - 2y^{(k-1)}, x^{(k)}; F]^{-1}$.
M^2_5 is outlined as

$$\begin{aligned} y^{(k)} &= x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}),\\ x^{(k+1)} &= y^{(k)} - [w^{(k)}, y^{(k)}; F]^{-1} F(y^{(k)}), \end{aligned}$$

where $w^{(k)} = x^{(k)} + B_9^{(k)} F(x^{(k)}),\ B_9^{(k)} = -\left(\frac{[3x^{(k)}-2y^{(k-1)},x^{(k)};F]+[2y^{(k-1)}-x^{(k)},x^{(k)};F]}{2}\right)^{-1}$.

Based on the given number of evaluations, the efficiency indices of M_p^i are denoted as E_p^i, while the computational costs are represented as C_p^i, as calculated from the formula (3.2). With these considerations, we can now reformulate the statement

$$
\begin{aligned}
C^1_{4.23607} &= \frac{m}{6}(2m^2 + 18m\mu + 21m + 21ml + 21l - 17) \text{ and } E^1_{4.23607} = 4.23607^{\frac{1}{C^1_{4.23607}}},\\
C^1_{4.44949} &= \frac{m}{3}(2m^2 + 12m\mu + 9m + 15ml + 9l - 5) \text{ and } E^1_{4.44949} = 4.44949^{\frac{1}{C^1_{4.44949}}},\\
C^1_{4.56155} &= \frac{m}{3}(2m^2 + 12m\mu + 9m + 15ml + 9l - 3\mu - 8) \text{ and } E^1_{4.56155} = 4.56155^{\frac{1}{C^1_{4.56155}}},\\
C^1_{5.23607} &= \frac{m}{3}(2m^2 + 12m\mu + 9m + 15ml + 9l - 5) \text{ and } E^1_{5.23607} = 5.23607^{\frac{1}{C^1_{5.23607}}},\\
C^1_{3} &= \frac{m}{3}(2m^2 + 6m\mu + 3m + 9ml + 3\mu + 3l - 2) \text{ and } E^1_{3} = 3^{\frac{1}{C^1_{3}}},\\
C^1_{5} &= \frac{m}{2}(2m^2 + 6m\mu + 3m + 9ml + 2\mu + 3l - 3) \text{ and } E^1_{5} = 5^{\frac{1}{C^1_{5}}},\\
C^1_{4.236} &= C^2_{4.236} = C^3_{4.236} = \frac{m}{2}(2m^2 + 6m\mu + 3m + 9ml + 2\mu + 3l - 3) \text{ and}\\
E^1_{4.236} &= E^2_{4.236} = E^3_{4.236} = 4.236^{\frac{1}{C^1_{4.236}}},\\
C^4_{4.236} &= \frac{m}{2}(2m^2 + 6m\mu + 3m + 9ml + 2\mu + 3l - 1) \text{ and } E^4_{4.236} = 4.236^{\frac{1}{C^4_{4.236}}},\\
C^2_{5} &= \frac{m}{2}(2m^2 + 8m\mu + 5m + 11ml + 2\mu + 3l - 1) \text{ and } E^2_{5} = 5^{\frac{1}{C^2_{5}}}.
\end{aligned}
$$

3.1 Efficiency comparison

To gauge the computational efficiency index between the iterative methods, such as M_p^i and M_q^j, we analyze the ratio

$$R^{i,j}_{p,q} = \frac{\log E^i_p}{\log E^j_q} = \frac{C^j_q \log p}{C^i_p \log q}.$$

When $R^{i,j}_{p,q} > 1$, it's evident that the iterative method M_p^i is more efficient than M_q^j. This holds true especially in scenarios where $\mu > 0$ and $l \geq 1$.

- $M^1_{4.23607}$ versus M^1_3

$$R^{1,1}_{4.23607,3} = \frac{\frac{m}{3}\left(2m^2 + 6m\mu + 3m + 9ml + 3\mu + 3l - 2\right)\log(4.23607)}{\frac{m}{6}\left(2m^2 + 18m\mu + 21m + 21ml + 21l - 17\right)\log(3)}.$$

  Here, the condition $R^{1,1}_{4.23607,3} > 1$ holds for $m \geq 4$, indicating that the efficiency index satisfies $E^1_{4.23607} > E^1_3$ for $m \geq 4$.
- $M^1_{4.44949}$ versus M^1_3

$$R^{1,1}_{4.44949,3} = \frac{\frac{m}{3}\left(2m^2 + 6m\mu + 3m + 9ml + 3\mu + 3l - 2\right)\log(4.44949)}{\frac{m}{3}\left(2m^2 + 12m\mu + 9m + 15ml + 9l - 5\right)\log(3)}.$$

  The inequality $R^{1,1}_{4.44949,3} > 1$ holds true when $m \geq 12$, indicating that $E^1_{4.44949} > E^1_3$ for $m \geq 12$.

- $M^1_{4.56155}$ versus M^1_3

$$R^{1,1}_{4.56155,3} = \frac{\frac{m}{3}\left(2m^2+6m\mu+3m+9ml+3\mu+3l-2\right)\log(4.56155)}{\frac{m}{3}\left(2m^2+12m\mu+9m+15ml+9l-3\mu-8\right)\log(3)}.$$

When $m \geq 10$, the inequality $R^{1,1}_{4.56155,3} > 1$ is valid, implying that $E^1_{4.56155} > E^1_3$ for $m \geq 10$.

- $M^1_{5.23607}$ versus M^1_3

$$R^{1,1}_{5.23607,3} = \frac{\frac{m}{3}\left(2m^2+6m\mu+3m+9ml+3\mu+3l-2\right)\log(5.23607)}{\frac{m}{3}\left(2m^2+12m\mu+9m+15ml+9l-5\right)\log(3)}.$$

When $m \geq 7$, the inequality $R^{1,1}_{5.23607,3} > 1$ is valid, confirming that the efficiency index $E^1_{5.23607}$ surpasses E^1_3.

- $M^1_{4.23607}$ versus M^1_5

$$R^{1,1}_{4.24,5} = \frac{\frac{m}{2}\left(2m^2+6m\mu+3m+9ml+2\mu+3l-3\right)\log(4.23607)}{\frac{m}{6}\left(2m^2+18m\mu+21m+21ml+21l-17\right)\log(5)}.$$

In this case, we have $R^{1,1}_{4.23607,5} > 1$ for $m \geq 4$, which means $E^1_{4.23607} > E^1_5$ for $m \geq 4$.

- $M^1_{4.44949}$ versus M^1_5

$$R^{1,1}_{4.44949,5} = \frac{\frac{m}{2}\left(2m^2+6m\mu+3m+9ml+2\mu+3l-3\right)\log(4.44949)}{\frac{m}{3}\left(2m^2+12m\mu+9m+15ml+9l-5\right)\log(5)}.$$

In this scenario, when $m \geq 10$, we find that $R^{1,1}_{4.44949,5} > 1$, indicating that $E^1_{4.44949} > E^1_5$.

- $M^1_{4.56155}$ versus M^1_5

$$R^{1,1}_{4.56155,5} = \frac{\frac{m}{2}\left(2m^2+6m\mu+3m+9ml+2\mu+3l-3\right)\log(4.56155)}{\frac{m}{3}\left(2m^2+12m\mu+9m+15ml+9l-3\mu-8\right)\log(5)}.$$

For $m \geq 9$, it follows that $R^{1,1}_{4.56155,5} > 1$, implying that the efficiency index $E^1_{4.56155}$ surpasses E^1_5.

- $M^1_{5.23607}$ versus M^1_5

$$R^{1,1}_{5.23607,5} = \frac{\frac{m}{2}\left(2m^2+6m\mu+3m+9ml+2\mu+3l-3\right)\log(5.23607)}{\frac{m}{3}\left(2m^2+12m\mu+9m+15ml+9l-5\right)\log(5)}.$$

It's a straightforward proof to show that $R^{1,1}_{5.23607,5} > 1$ for $m \geq 6$. Thus, we can conclude $E^1_{5.23607} > E^1_5$ for $m \geq 6$.

- $M^1_{4.23607}$ versus $M^1_{4.236}$

$$R^{1,1}_{4.23607,4.236} = \frac{\frac{m}{2}\left(2m^2+6m\mu+3m+9ml+2\mu+3l-3\right)\log(4.23607)}{\frac{m}{6}\left(2m^2+18m\mu+21m+21ml+21l-17\right)\log(4.236)}.$$

It's apparent that for $m \geq 2$, the ratio $R^{1,1}_{4.23607,4.236} > 1$. This implies that the efficiency index of $M^1_{4.23607}$ is greater than that of $M^1_{4.236}$. Moreover, considering that the computational cost and order of methods $M^2_{4.236}$ and $M^3_{4.236}$ are the same as $M^1_{4.236}$, we can infer that for $m \geq 2$, $E^1_{4.23607}$ is also greater than $E^2_{4.236}$ and $E^3_{4.236}$.

- $M^1_{4.44949}$ versus $M^1_{4.236}$

$$R^{1,1}_{4.44949,4.236} = \frac{\frac{m}{2}\left(2m^2+6m\mu+3m+9ml+2\mu+3l-3\right)\log(4.44949)}{\frac{m}{3}\left(2m^2+12m\mu+9m+15ml+9l-5\right)\log(4.236)}.$$

It's evident that $R^{1,1}_{4.44949,4.236} > 1$ for $m \geq 6$. This indicates that $E^1_{4.44949} > E^1_{4.236}$ for $m \geq 6$. Also for $m \geq 6$, indicating $E^1_{4.44949} > E^2_{4.236}$ and $E^1_{4.44949} > E^3_{4.236}$.

- $M^1_{4.56155}$ versus $M^1_{4.236}$

$$R^{1,1}_{4.56155,4.236} = \frac{\frac{m}{2}\left(2m^2 + 6m\mu + 3m + 9ml + 2\mu + 3l - 3\right)\log(4.56155)}{\frac{m}{3}\left(2m^2 + 12m\mu + 9m + 15ml + 9l - 3\mu - 8\right)\log(4.236)}.$$

It's clear that $R^{1,1}_{4.56155,4.236} > 1$ for $m \geq 5$, indicating that $E^1_{4.56155} > E^1_{4.236}$ for $m \geq 5$. Additionally, for $m \geq 5$, we can infer that $E^1_{4.56155} > E^2_{4.236}$ and $E^1_{4.56155} > E^3_{4.236}$.

- $M^1_{5.23607}$ versus $M^1_{4.236}$

$$R^{1,1}_{5.23607,4.236} = \frac{\frac{m}{2}\left(2m^2 + 6m\mu + 3m + 9ml + 2\mu + 3l - 3\right)\log(5.23607)}{\frac{m}{3}\left(2m^2 + 12m\mu + 9m + 15ml + 9l - 5\right)\log(4.236)}.$$

For $m \geq 4$, it's observable that $R^{1,1}_{5.23607,4.236} > 1$. This suggests that $E^1_{5.23607} > E^1_{4.236}$. Similarly, for $m \geq 4$, it implies that $E^1_{5.23607} > E^2_{4.236}$ and $E^1_{5.23607} > E^3_{4.236}$.

- $M^1_{4.23607}$ versus $M^4_{4.236}$

$$R^{1,4}_{4.23607,4.236} = \frac{\frac{m}{2}\left(2m^2 + 6m\mu + 3m + 9ml + 2\mu + 3l - 1\right)\log(4.23607)}{\frac{m}{6}\left(2m^2 + 18m\mu + 21m + 21ml + 21l - 17\right)\log(4.236)}.$$

We can readily demonstrate that $R^{1,4}_{4.23607,4.236} > 1$ holds true for $m \geq 1$. Consequently, we infer that $E^1_{4.23607} > E^4_{4.236}$.

- $M^1_{4.44949}$ versus $M^4_{4.236}$

$$R^{1,4}_{4.44949,4.236} = \frac{\frac{m}{2}\left(2m^2 + 6m\mu + 3m + 9ml + 2\mu + 3l - 1\right)\log(4.44949)}{\frac{m}{3}\left(2m^2 + 12m\mu + 9m + 15ml + 9l - 5\right)\log(4.236)}.$$

It's straightforward to demonstrate that $R^{1,4}_{4.44949,4.236} > 1$ for $m \geq 6$. Consequently, we deduce that $E^1_{4.44949} > E^4_{4.236}$.

- $M^1_{4.56155}$ versus $M^4_{4.236}$

$$R^{1,4}_{4.56155,4.236} = \frac{\frac{m}{2}\left(2m^2 + 6m\mu + 3m + 9ml + 2\mu + 3l - 1\right)\log(4.56155)}{\frac{m}{3}\left(2m^2 + 12m\mu + 9m + 15ml + 9l - 3\mu - 8\right)\log(4.236)}.$$

It's evident from our analysis that for $m \geq 4$, $R^{1,4}_{4.56155,4.236} > 1$. Therefore, we infer that $E^1_{4.56155} > E^4_{4.236}$.

- $M^1_{5.23607}$ versus $M^4_{4.236}$

$$R^{1,4}_{5.23607,4.236} = \frac{\frac{m}{2}\left(2m^2 + 6m\mu + 3m + 9ml + 2\mu + 3l - 1\right)\log(5.23607)}{\frac{m}{3}\left(2m^2 + 12m\mu + 9m + 15ml + 9l - 5\right)\log(4.236)}.$$

Establishing that $R^{1,4}_{5.23607,4.236} > 1$ for $m \geq 3$ is a straightforward endeavor. Consequently, we can deduce that $E^1_{5.23607} > E^4_{4.236}$.

- $M^1_{4.23607}$ versus M^2_5

$$R^{1,2}_{4.23607,5} = \frac{\frac{m}{2}\left(2m^2 + 8m\mu + 5m + 11ml + 2\mu + 3l - 1\right)\log(4.23607)}{\frac{m}{6}\left(2m^2 + 18m\mu + 21m + 21ml + 21l - 17\right)\log(5)}.$$

It can be readily confirmed that for $m \geq 1$, $R^{1,2}_{4.23607,5} > 1$, signifying that $E^1_{4.23607} > E^2_5$.

- $M^1_{4.44949}$ versus M^2_5

$$R^{1,2}_{4.44949,5} = \frac{\frac{m}{2}\left(2m^2 + 8m\mu + 5m + 11ml + 2\mu + 3l - 1\right)\log(4.44949)}{\frac{m}{3}\left(2m^2 + 12m\mu + 9m + 15ml + 9l - 5\right)\log(5)}.$$

It's easily verifiable that for $m \geq 3$, $R^{1,2}_{4.44949,5} > 1$, indicating that $E^1_{4.44949} > E^2_5$.

- $M^1_{4.56155}$ versus M^2_5

$$R^{1,2}_{4.56155,5} = \frac{\frac{m}{2}\left(2m^2 + 8m\mu + 5m + 11ml + 2\mu + 3l - 1\right)\log(4.56155)}{\frac{m}{3}\left(2m^2 + 12m\mu + 9m + 15ml - 3\mu + 9l - 8\right)\log(5)}.$$

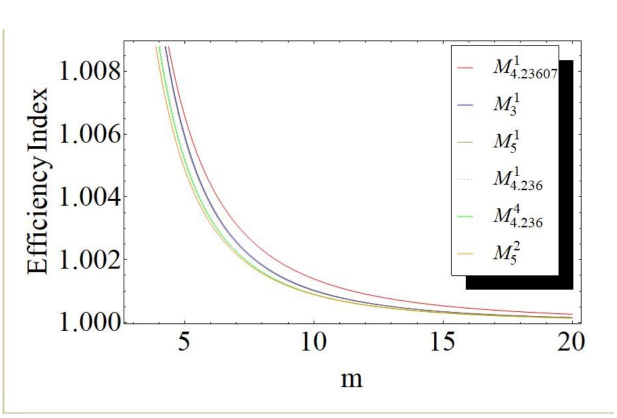
For $m \geq 1$, it's clear that $R^{1,2}_{4.56155,5} > 1$, indicating that $E^1_{4.56155} > E^2_5$.

- $M^1_{5.23607}$ versus M^2_5

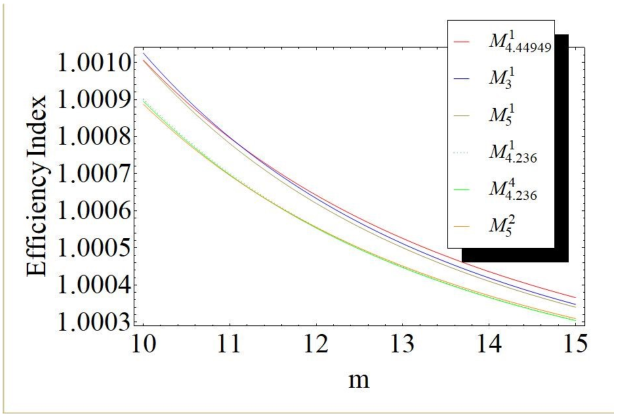
$$R^{1,2}_{5.23607,5} = \frac{\frac{m}{2}\left(2m^2 + 8m\mu + 5m + 11ml + 2\mu + 3l - 1\right)\log(5.23607)}{\frac{m}{3}\left(2m^2 + 12m\mu + 9m + 15ml + 9l - 5\right)\log(5)}.$$

For $m \geq 1$, it's apparent that $R^{1,2}_{5.23607,5} > 1$, implying that $E^1_{5.23607} > E^2_5$.

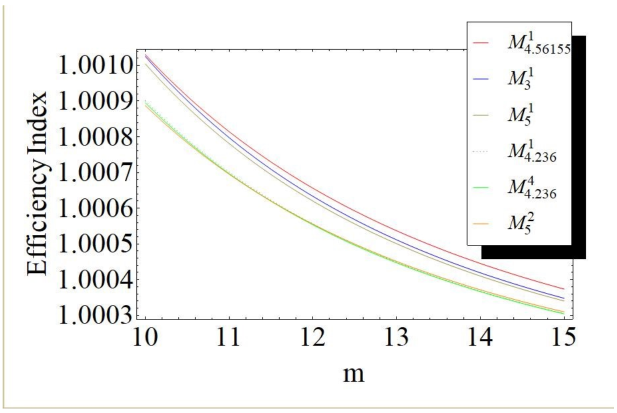
The previously discussed results are graphically represented in Figure 3.1.



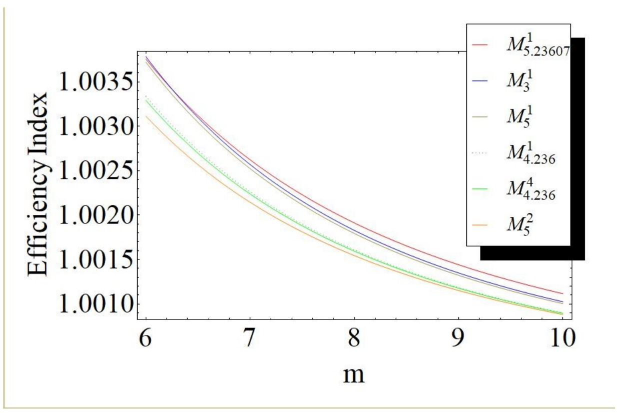
(a) Plots for efficiency index values for $M^1_{4.23607}$ versus M^1_3, M^1_5, $M^1_{4.236}$, $M^4_{4.236}$, M^2_5



(b) Plots for efficiency index values for $M^1_{4.44949}$ versus M^1_3, M^1_5, $M^1_{4.236}$, $M^4_{4.236}$, M^2_5



(c) Plots for efficiency index values for $M^1_{4.56155}$ versus M^1_3, M^1_5, $M^1_{4.236}$, $M^4_{4.236}$ and M^2_5



(d) Plots for efficiency index values for $M^1_{5.23607}$ versus M^1_3, M^1_5, $M^1_{4.236}$, $M^4_{4.236}$ and M^2_5

Fig. 3.1: Comparative plots for efficiency index values of different methods

Figure 3.1 presents a comparative analysis of the proposed schemes, specifically $M^1_{4.23607}$, $M^1_{4.44949}$, $M^1_{4.56155}$, $M^1_{5.23607}$. The findings clearly indicate that these methods consistently outperform the well-established existing approaches, such as M^1_3, M^1_5, $M^1_{4.236}$, $M^4_{4.236}$, and M^2_5. This improved efficiency is consistently observed across all evaluated scenarios.

4 Numerical results

In this section, we address several numerical problems to demonstrate the convergence behavior of the proposed methods. We compare our method with existing methods M_3^1 and M_5^1 in the given ref. [4], as well as $M_{4.236}^1$, $M_{4.236}^2$, $M_{4.236}^3$, $M_{4.236}^4$, and M_5^2 as documented in ref. [18]. The computations are conducted in Mathematica 8.0 [19] using multiple-precision arithmetic, set to 2048 digits to ensure high accuracy. To confirm the theoretical convergence rate, we compute the computational order of convergence (p_c) using the following formula [13]

$$p_c = \frac{\log(\|F(x^{(k)})\|/\|F(x^{(k-1)})\|)}{\log(\|F(x^{(k-1)})\|/\|F(x^{(k-2)})\|)}.$$

This calculation incorporates the last four iterations to determine the convergence rate accurately. For all proposed and existing methods including M_3^1, M_5^1, $M_{4.236}^1$, $M_{4.236}^2$, $M_{4.236}^3$, $M_{4.236}^4$ and M_5^2, we initialize $B_i^{(k)}$ with $B_1^{(0)} = B_2^{(0)} = B_3^{(0)} = B_4^{(0)} = B_5^{(0)} = B_6^{(0)} = B_7^{(0)} = B_8^{(0)} = B_9^{(0)} = -0.01I$, where I is the identity matrix.

Example 4.1. Consider the following system of two equations (see [6])

$$\begin{cases} x_1^2 - x_1 - x_2^2 - 1 = 0, \\ x_2 - \sin x_1 = 0. \end{cases}$$

with initial guess $x^{(0)} = \{2, 2\}^\top$. The solution to this system is given by

$$\alpha = \{1.952913098702211788\cdots, 0.927877401589489631\cdots\}^\top.$$

Table 4.1: Numerical results for Example 4.1

Methods	$\|x^{(1)} - \alpha\|$	$\|x^{(2)} - \alpha\|$	$\|x^{(3)} - \alpha\|$	$\|x^{(4)} - \alpha\|$	p_c	**CPU**
M_3^1	0.259	0.00662	8.69×10^{-8}	2.48×10^{-21}	2.710	0.04350
M_5^1	0.0427	1.20×10^{-8}	4.38×10^{-34}	4.63×10^{-137}	4.0963	0.04866
$M_{4.236}^1$	0.0427	2.13×10^{-6}	6.64×10^{-26}	3.72×10^{-109}	4.2265	0.03124
$M_{4.236}^2$	0.0427	6.37×10^{-6}	1.21×10^{-23}	2.85×10^{-99}	4.2229	0.03128
$M_{4.236}^3$	0.0427	1.12×10^{-7}	1.26×10^{-31}	6.06×10^{-134}	4.2370	0.04687
$M_{4.236}^4$	0.0427	4.66×10^{-7}	1.51×10^{-28}	2.07×10^{-120}	4.2350	0.04683
M_5^2	0.0427	1.68×10^{-8}	9.87×10^{-42}	5.73×10^{-209}	4.9923	0.05958
$M_{4.23607}^1$	0.00367	7.94×10^{-12}	5.70×10^{-49}	6.98×10^{-207}	4.2433	0.03120
$M_{4.44949}^1$	0.00367	5.67×10^{-14}	1.66×10^{-61}	4.15×10^{-274}	4.4525	0.03120
$M_{4.56155}^1$	0.00367	2.64×10^{-12}	1.76×10^{-54}	1.35×10^{-247}	4.5686	0.03121
$M_{5.23607}^1$	0.00367	1.12×10^{-13}	4.59×10^{-67}	1.34×10^{-334}	5.0039	0.03121

Example 4.2. Now, considering the system of six equations as described in the ref. [16]

$$\sum_{j=1, j\neq i}^{6} x_j - e^{-x_i} = 0, 1 \leq i \leq 6$$

with initial guess $x^{(0)} = \{-0.1, -0.1, -0.1, -0.1, -0.1, -0.1\}^\top$. The solution of this problem α is

$$\begin{Bmatrix} 0.168915973499109565\cdots, 0.168915973499109565\cdots, 0.168915973499109565\cdots, \\ 0.168915973499109565\cdots, 0.168915973499109565\cdots, 0.168915973499109565\cdots \end{Bmatrix}^\top.$$

Table 4.2: Numerical results for Example 4.2

Methods	$\Vert x^{(1)}-\alpha\Vert$	$\Vert x^{(2)}-\alpha\Vert$	$\Vert x^{(3)}-\alpha\Vert$	$\Vert x^{(4)}-\alpha\Vert$	p_c	CPU
M_3^1	0.0138	1.17×10^{-8}	3.05×10^{-24}	1.50×10^{-67}	2.779	0.34300
M_5^1	0.000269	5.19×10^{-20}	1.03×10^{-80}	2.78×10^{-323}	4.0219	0.45021
$M_{4.236}^1$	0.000269	1.10×10^{-20}	1.19×10^{-89}	6.65×10^{-382}	4.2376	0.49723
$M_{4.236}^2$	0.000269	2.46×10^{-20}	6.69×10^{-88}	3.35×10^{-374}	4.2374	0.48420
$M_{4.236}^3$	0.000269	2.57×10^{-21}	8.82×10^{-93}	1.18×10^{-395}	4.2382	0.49541
$M_{4.236}^4$	0.000269	1.04×10^{-20}	9.68×10^{-90}	2.78×10^{-382}	4.2377	0.51674
M_5^2	0.000269	1.99×10^{-24}	2.39×10^{-125}	5.85×10^{-630}	5.0000	0.56555
$M_{4.23607}^1$	5.48×10^{-6}	8.91×10^{-28}	2.73×10^{-120}	3.73×10^{-512}	4.2357	0.30950
$M_{4.44949}^1$	5.48×10^{-6}	4.02×10^{-29}	8.05×10^{-132}	6.97×10^{-589}	4.4505	0.34371
$M_{4.56155}^1$	5.48×10^{-6}	9.37×10^{-30}	2.10×10^{-138}	4.06×10^{-634}	4.5625	0.37492
$M_{5.23607}^1$	5.48×10^{-6}	2.12×10^{-32}	3.28×10^{-165}	2.87×10^{-829}	5.0000	0.49408

Example 4.3. Now, considering the system of seven equations (taken from [3])

$$x_i-\cos\left(2x_i-\sum_{j=1}^{4}x_j\right)=0, 1\le i\le 7$$

with initial guess $x^{(0)}=\{0.7,0.7,\cdots,0.7\}^\top$. The solution of this problem is

$$\alpha=\{0.514933264661129425...,0.514933264661129425...,.....,0.514933264661129425...\}^\top.$$

Table 4.3: Numerical results for Example 4.3

Methods	$\Vert x^{(1)}-\alpha\Vert$	$\Vert x^{(2)}-\alpha\Vert$	$\Vert x^{(3)}-\alpha\Vert$	$\Vert x^{(4)}-\alpha\Vert$	p_c	CPU
M_3^1	0.0173	4.72×10^{-7}	5.80×10^{-19}	6.48×10^{-52}	2.767	0.23766
M_5^1	0.000877	2.85×10^{-15}	1.02×10^{-60}	5.25×10^{-243}	4.0111	0.42459
$M_{4.236}^1$	0.000877	3.12×10^{-16}	1.41×10^{-68}	2.09×10^{-290}	4.2378	0.49129
$M_{4.236}^2$	0.000877	6.84×10^{-16}	7.31×10^{-67}	7.42×10^{-283}	4.2375	0.41468
$M_{4.236}^3$	0.000877	7.02×10^{-17}	9.00×10^{-72}	1.95×10^{-304}	4.2386	0.40035
$M_{4.236}^4$	0.000877	3.02×10^{-16}	1.24×10^{-68}	1.20×10^{-290}	4.2379	0.46973
M_5^2	0.000877	5.07×10^{-20}	1.41×10^{-100}	2.36×10^{-503}	5.0000	0.55258
$M_{4.23607}^1$	0.000363	6.62×10^{-18}	3.20×10^{-76}	2.92×10^{-323}	4.2362	0.36387
$M_{4.44949}^1$	0.000363	6.74×10^{-19}	4.81×10^{-84}	4.32×10^{-374}	4.4522	0.35929
$M_{4.56155}^1$	0.000363	1.27×10^{-19}	5.13×10^{-90}	4.54×10^{-411}	4.5608	0.37488
$M_{5.23607}^1$	0.000363	1.77×10^{-21}	7.30×10^{-108}	8.78×10^{-540}	5.0000	0.35929

Example 4.4. Now, considering the system of ten equations (see [18])

$$1-2\sum_{j=1,j\neq i}^{10}x_j^2+\arctan x_i=0, 1\le i\le 10$$

with initial guess $x^{(0)}=\{0.3,0.3,\cdots,0.3\}^\top$. The solution of this problem is

$$\alpha=\{0.264417769843922183...,0.264417769843922183...,.....,0.264417769843922183...\}^\top.$$

Table 4.4: Numerical results for Example 4.4

Methods	$\|x^{(1)}-\alpha\|$	$\|x^{(2)}-\alpha\|$	$\|x^{(3)}-\alpha\|$	$\|x^{(4)}-\alpha\|$	p_c	**CPU**
M_3^1	0.00803	2.31×10^{-7}	1.13×10^{-20}	1.14×10^{-56}	2.704	9.20833
M_5^1	0.000609	1.97×10^{-17}	3.00×10^{-69}	1.17×10^{-278}	4.0413	13.58254
$M_{4.236}^1$	0.000609	1.07×10^{-14}	6.58×10^{-60}	1.63×10^{-251}	4.2379	14.68128
$M_{4.236}^2$	0.000609	2.33×10^{-14}	3.28×10^{-58}	4.93×10^{-244}	4.2376	14.59590
$M_{4.236}^3$	0.000609	2.15×10^{-15}	2.64×10^{-63}	2.12×10^{-266}	4.2390	14.69543
$M_{4.236}^4$	0.000609	9.47×10^{-15}	3.99×10^{-60}	1.96×10^{-252}	4.2382	14.14806
M_5^2	0.000609	1.71×10^{-17}	2.96×10^{-85}	4.62×10^{-424}	5.0000	18.78451
$M_{4.23607}^1$	0.000172	5.16×10^{-17}	2.82×10^{-70}	6.67×10^{-296}	4.2362	9.27908
$M_{4.44949}^1$	0.000172	3.90×10^{-20}	5.18×10^{-89}	8.33×10^{-396}	4.4543	11.77912
$M_{4.56155}^1$	0.000172	1.44×10^{-18}	9.39×10^{-83}	1.58×10^{-375}	4.5613	12.18464
$M_{5.23607}^1$	0.000172	3.06×10^{-20}	5.50×10^{-99}	1.02×10^{-492}	5.0000	12.21653

Example 4.5. Consider the boundary value problem (see [12])

$$y'' = \frac{1}{2}y^3 + 3y' - \frac{3}{2-x} + \frac{1}{2}, \quad y(0) = 0, \quad y(1) = 1.$$

Discretizing the interval $[0,1]$ with a uniform partition

$$x_0 = 0 < x_1 < x_2 < \cdots < x_{n-1} < x_n = 1, \quad x_{i+1} = x_i + h, \quad h = \frac{1}{n},$$

we define the discrete values $y_k \approx y(x_k)$ for $k = 0,1,\ldots,n$. Using central difference approximations for the first and second derivatives

$$y'_k \approx \frac{y_{k+1} - y_{k-1}}{2h}, \quad y''_k \approx \frac{y_{k-1} - 2y_k + y_{k+1}}{h^2}, \quad k = 1,2,\ldots,n-1,$$

we obtain the discrete system of nonlinear equations

$$y_{k+1} - 2y_k + y_{k-1} - \frac{h^2}{2}y_k^3 - \frac{3}{2}h(y_{k+1} - y_{k-1}) + \frac{3}{2-x_k}h^2 - \frac{1}{2}h^2 = 0, \quad k = 1,2,\ldots,n-1.$$

For $n = 5$, we solve this system using the initial guess $y^{(0)} = (1,1,1,1)^\top$, yielding the approximate solution

$$\alpha = \{0.1100327722..., 0.2477507450..., 0.4253757529..., 0.6635723429...\}^\top.$$

Table 4.5: Numerical results for Example 4.5

Methods	$\|x^{(1)}-\alpha\|$	$\|x^{(2)}-\alpha\|$	$\|x^{(3)}-\alpha\|$	$\|x^{(4)}-\alpha\|$	p_c	**CPU**
M_3^1	0.0892	4.36×10^{-6}	5.30×10^{-17}	2.04×10^{-47}	2.8000	0.04686
M_5^1	0.00426	2.04×10^{-14}	3.76×10^{-58}	1.37×10^{-234}	4.0381	0.07811
$M_{4.236}^1$	0.00426	1.42×10^{-15}	5.18×10^{-68}	3.40×10^{-290}	4.2413	0.04690
$M_{4.236}^2$	0.00426	3.21×10^{-15}	3.04×10^{-66}	2.06×10^{-282}	4.2410	0.06248
$M_{4.236}^3$	0.00426	2.39×10^{-16}	1.04×10^{-71}	2.28×10^{-306}	4.2423	0.04687
$M_{4.236}^4$	0.00426	9.25×10^{-16}	9.34×10^{-69}	2.35×10^{-293}	4.2419	0.04686
M_5^2	0.00426	1.43×10^{-17}	4.84×10^{-91}	2.22×10^{-458}	5.0045	0.06249
$M_{4.23607}^1$	0.00035	2.39×10^{-20}	4.04×10^{-89}	4.22×10^{-380}	4.2318	0.03124

Methods	$\|x^{(1)}-\alpha\|$	$\|x^{(2)}-\alpha\|$	$\|x^{(3)}-\alpha\|$	$\|x^{(4)}-\alpha\|$	p_c	CPU
$M^1_{4.44949}$	0.00035	2.13×10^{-22}	1.16×10^{-101}	2.55×10^{-454}	4.4329	0.04681
$M^1_{4.56155}$	0.00035	1.03×10^{-22}	2.16×10^{-106}	4.50×10^{-488}	4.5652	0.04683
$M^1_{5.23607}$	0.00035	6.29×10^{-24}	5.72×10^{-23}	4.81×10^{-618}	5.0024	0.04682

Example 4.6. Consider a system of nonlinear equations in two variable which involved both differentiable and non-differentiable components. These equations can be effectively solved only by employing derivative-free iterative methods (see [15])

$$\begin{cases} x_1^2 - x_2 + 1 + \frac{1}{9}|x_1 - 1| = 0, \\ x_2^2 + x1 - 7 + \frac{1}{9}|x_2| = 0. \end{cases}$$

with initial guess $x^{(0)} = \{3, 3\}^\top$. The solution to this system is given by

$$\alpha = \{1.159360850193451399\cdots, 2.3618243420938881695\cdots\}^\top.$$

Table 4.6: Numerical results for Example 4.6

Methods	$\|x^{(1)}-\alpha\|$	$\|x^{(2)}-\alpha\|$	$\|x^{(3)}-\alpha\|$	$\|x^{(4)}-\alpha\|$	p_c	CPU
M^1_3	0.535	0.0119	1.28×10^{-6}	2.99×10^{-19}	3.1820	0.02304
M^1_5	0.195	1.26×10^{-6}	1.43×10^{-23}	1.18×10^{-116}	5.4933	0.03008
$M^1_{4.236}$	0.195	5.28×10^{-5}	1.11×10^{-19}	6.30×10^{-82}	4.2404	0.02756
$M^2_{4.236}$	0.195	8.48×10^{-5}	1.62×10^{-18}	1.06×10^{-76}	4.2418	0.03210
$M^3_{4.236}$	0.195	3.05×10^{-6}	3.20×10^{-25}	6.36×10^{-106}	4.2520	0.02881
$M^4_{4.236}$	0.195	7.36×10^{-5}	4.47×10^{-19}	2.33×10^{-79}	4.2403	0.02746
M^2_5	0.195	2.12×10^{-6}	8.48×10^{-31}	8.74×10^{-153}	5.0000	0.02695
$M^1_{4.23607}$	0.157	3.96×10^{-5}	1.02×10^{-20}	8.68×10^{-87}	4.2470	0.02233
$M^1_{4.44949}$	0.157	6.13×10^{-7}	1.72×10^{-33}	3.02×10^{-166}	5.0001	0.02669
$M^1_{4.56155}$	0.157	6.02×10^{-6}	3.61×10^{-26}	2.37×10^{-118}	4.5584	0.02588
$M^1_{5.23607}$	0.157	1.33×10^{-6}	8.45×10^{-32}	8.54×10^{-158}	5.0001	0.02236

Figure 4.1 illustrates the error progression at each iteration for the various iterative methods used in solving Examples 1–6. The graph demonstrates the superior efficiency of our proposed method (2.3)–(2.6) when compared to other established approaches. Notably, it shows that our method converges faster and attains considerably greater computational accuracy in all the examples.

In addition to achieving high accuracy, computational efficiency is a crucial factor in the evaluation of iterative methods. For the six examined examples, our proposed method demonstrates a significant reduction in CPU time compared to existing approaches, as shown in Figure 4.2. This advantage is attributed to the optimized structure of our iterative scheme. As a result, our method not only delivers superior accuracy but also enhances overall computational performance, making it a highly effective choice for solving nonlinear systems.

These findings collectively emphasize the benefits of our approach in terms of both precision and efficiency, particularly when compared to other widely recognized methods in the literature.

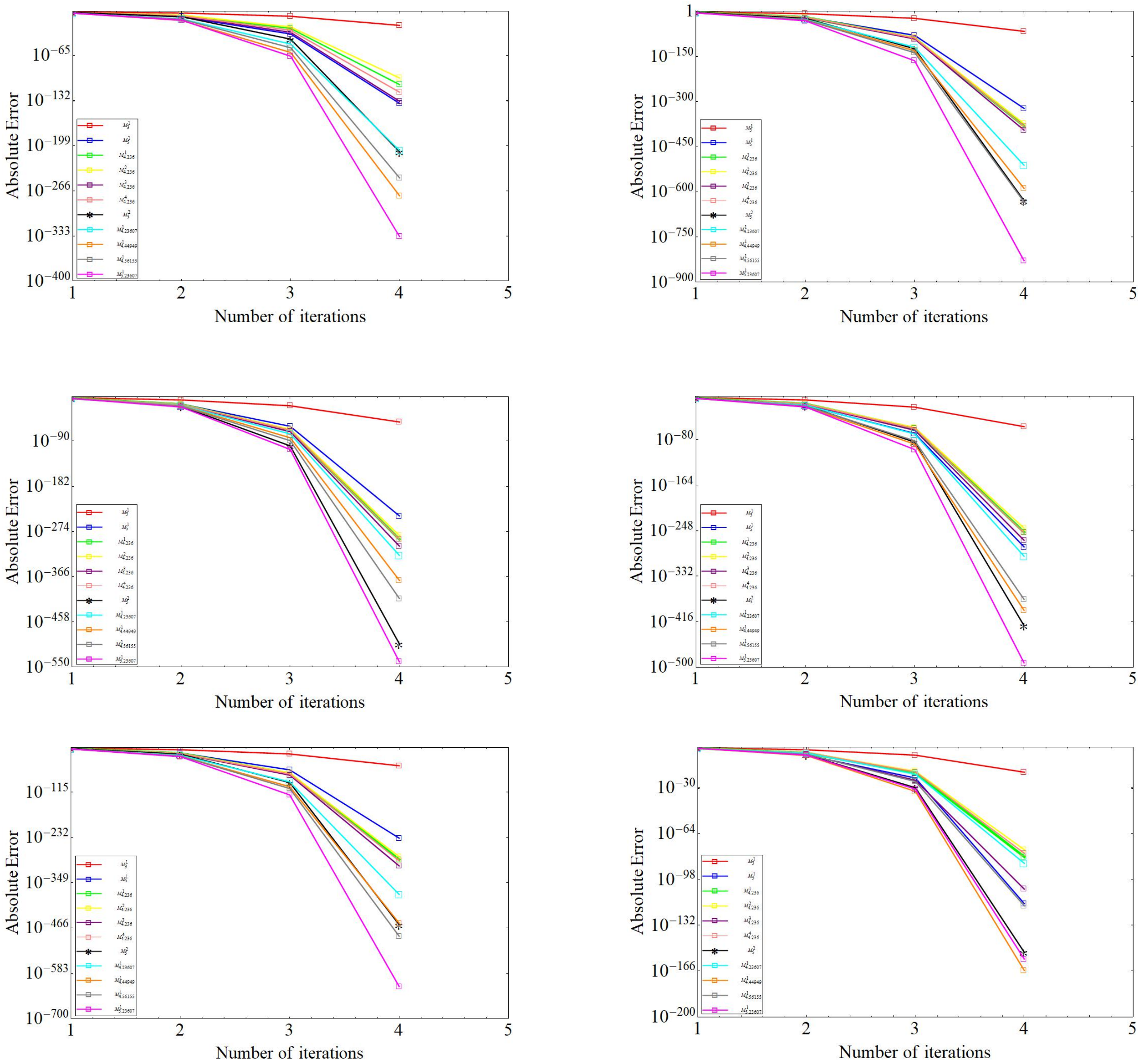


Fig. 4.1: Graphical analysis of the absolute error for Examples 4.1 to 4.6.

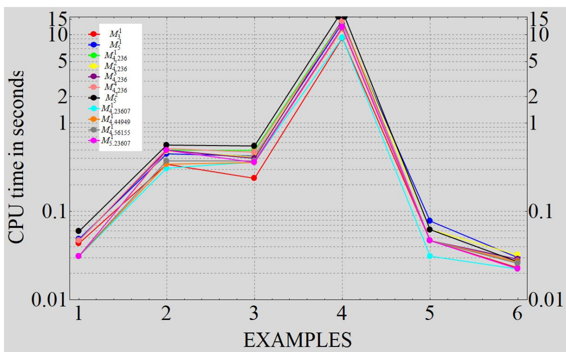


Fig. 4.2: Comparison of graphs for different methods applied to each example based on CPU time.

5 Conclusion

We've introduced a class of two-point derivative-free iterative methods with memory for solving systems of nonlinear equations. By employing the first-order divided difference operator for functions of multiple variables and Taylor's expansion, we've estimated the lower bound of the R-order of convergence for these new methods. It's been established that their convergence order is 4.23607, 4.44949, 4.56155 and 5.23607. To accelerate convergence, we utilize a self-accelerating parameter in the form of a self-corrected matrix as the iteration progresses, paving the way for improving the convergence speed of methods for solving systems of nonlinear equations. Additionally, we've compared the computational efficiency of the new methods with that of some existing methods. The comparison, along with the outcomes of numerical experiments indicates that our presented methods outperform similar existing ones. These numerical results validate the robust and efficient nature of the proposed techniques.

6 Acknowledgment

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).

Declarations

Conflict of Interest.
The authors declare that there is no conflict of interest.
Funding.
This research received no external funding.
Author Contributions.
All the authors contributed equally to the research and preparation of this article.

References

1. Argyros, I.K.: Advances in the Efficiency of Computational Methods and Applications. World Scientific, Singapore (2000)
2. Behl, R., Cordero, A., Torregrosa, J.R., Bhalla, S.: A new high-order Jacobian-free iterative method with memory for solving nonlinear systems. Mathematics. **9**, 1-16 (2021)
3. Capdevila, R.R., Cordero, A., Torregrosa, J.R.: A new three-step class of iterative methods for solving nonlinear systems. Mathematics. **7** (12), 1221 (2019)
4. Chicharro, F.I., Cordero, A., Garrido, N., Torregrosa, J.R.: On the improvement of the order of convergence of iterative methods for solving nonlinear systems by means of memory. Appl. Math. Lett. **104**, 106277 (2020)
5. Cordero, A., Garrido, N., Torregrosa, J.R., Triguero, P.N.: Design of iterative methods with memory for solving nonlinear systems. Math. Methods Appl. Sci. **46**, 12361–12377 (2023)
6. Cordero, A., Maimo, J.G., Torregrosa, J.R., Vassileva, M.P.: Iterative methods with memory for solving systems of nonlinear equations using a second order approximation. Mathematics. **7** (11), 1069 (2019)
7. Cordero, A., Garrido, N., Torregrosa, J.R., Triguero, P.N.: Improving the order of a fifth-order family of vectorial fixed point schemes by introducing memory. Fixed Point Theory. **24**, 155-172 (2023)

8. Grau-Sánchez, M., Grau, Á., Noguera, M.: Ostrowski type methods for solving systems of nonlinear equations. Appl. Math. Comput. **218**(6), 2377-2385 (2011)
9. Grau-Sánchez, M., Noguera, M.: A technique to choose the most efficient method between Secant method and some variants. Appl. Math. Comput. **218** (11), 6415-6426 (2012)
10. Grau-Sánchez, M., Grau, Á., Noguera, M.: Frozen divided difference scheme for solving systems of nonlinear equations. J. Comput. Appl. Math. **235** (6), 1739-1743 (2011)
11. Grau-Sánchez, M., Noguera, M., Amat, S.: On the approximation of derivatives using divided difference operators preserving the local convergence order of iterative methods. J. Comput. Appl. Math. **237** (1), 363-372 (2013)
12. Ortega, J.M., Rheinboldt, W.C.: Iterative solution of nonlinear equations in several variables. SIAM, 289 (2000)
13. Petkovic, M. S.: On a general class of multipoint root-finding methods of high computational efficiency. SIAM Journal on Numerical Analysis. **47** (6), 4402-4414 (2010)
14. Petkovic, M. S., Sharma, J. R.: On some efficient derivative-free iterative methods with memory for solving systems of nonlinear equations. Numer. Algorithms. **71**, 457-474 (2016)
15. Shakhno, S., Yarmola, H.: On the two-step method for solving nonlinear equations with non differentiable operator. PAMM. **12** (1), 617-618 (2012)
16. Sharma, J. R.; Arora, H.: An efficient derivative-free iterative method for solving systems of nonlinear equations. Appl. Anal. Discrete Math. 390-403 (2013)
17. Steffensen, J.: Remarks on iteration. Scandinavian Actuarial Journal. **1**, 64-72 (1993)
18. Wang, X.; Jin, Y.; Zhao, Y.: Derivative-free iterative methods with some Kurchatov-type accelerating parameters for solving nonlinear systems. Symmetry. **13** (6), 943 (2021)
19. Wolfram, S.: The Mathematica Book. Wolfram Research, Inc. (2003)

Received 21.03.2025
Revised 28.05.2025